Fast Two-Robot Disk Evacuation with Wireless Communication*

Ioannis Lamprou(✉),†, Russell Martin†, Sven Schewe‡
{Forename.Surname}@liverpool.ac.uk

Department of Computer Science
University of Liverpool, UK

October 28, 2019

Abstract

In the fast evacuation problem, we study the path planning problem for two robots who want to minimize the worst-case evacuation time on the unit disk, that is, the time till both of them evacuate. The robots are initially placed at the center of the disk. In order to evacuate, they need to reach an unknown point, the exit, on the boundary of the disk. Once one of the robots finds the exit, it will instantaneously, i.e., using wireless communication, notify the other agent who will then follow a straight line to it.

The problem has been studied for robots with the same speed [13]. We study a more general case where one robot has speed 1 and the other has speed $s \geq 1$. We provide optimal evacuation strategies in the case that $s \geq 2.75$ by showing matching upper and lower bounds on the worst-case evacuation time. For $1 \leq s < 2.75$, we show (non-matching) upper and lower bounds on the evacuation time with a ratio less than 1.22. Moreover, we demonstrate that a different-speeds generalization of the two-robot search strategy from [13] is outperformed by our proposed strategies for any $s \geq 1.71$.

Keywords. Evacuation; Different Speeds; Disk; Wireless; Fast Robots

1 Introduction

Consider a pair of mobile robots in an environment represented by a circular disk of unit radius. The goal of the robots is to find an *exit*, i.e., a point at an unknown location on the boundary of the disk, and both move to this exit. The exit is only recognized when a robot visits it. The robots' aim is to accomplish this task as quickly as possible. This problem is referred to as the *evacuation problem*. The robots start at the center of the disk and can move with a speed not exceeding their maximum velocity, which may be different from one another. They can coordinate their actions in any manner they like, and can communicate wirelessly (instantaneously).

1.1 Related work

Evacuation belongs to the realm of distributed search problems, which have a long history in mathematics, computer science, and operations research, see, e.g., [4, 5, 6].

Salient features in search problems include the *environment* (a geometric one or graph-based), *mobility* of the robots (how they are allowed to move), *perception* of and *interaction* with the environment, and their *computational* and *communication abilities*. Typical tasks include exploring and mapping an unknown environment or finding a (mobile or immobile) target. Examples include cops and robbers games [7] and

*A preliminary version of the article appeared in DISC 2016, LNCS 9888, pp. 1–15, Springer, Heidelberg, 2016.

†Partially supported by the Network Sciences and Technologies (NeST) initiative, EEE & CS, University of Liverpool.

‡Partially supported by EPSRC grants EP/P020909/1 "Solving Parity Games in Theory and Practice" and EP/M027287/1 "Energy Efficient Control".

pursuit-evasion games [30], the "lost at sea" problem [24], the cow-path problem and the plane-searching problem [3, 8, 26, 27]. Other tasks are rendezvous or gathering of mobile agents [28, 29] and evacuation [11, 13, 18]. (Note that we distinguish between the distributed version of evacuation problems involving a search for an unknown exit, and centralized versions typically modeled as (dynamic) capacitated flow problems on graphs, where the exit is known.) General surveys on search and rendezvous problems can be found in [1, 2, 21]. Another related problem is the task of patrolling or monitoring, i.e., the periodic (re)visitation of (part of) the environment [10, 14, 32].

In most of these settings, the typical cost is the time required to finish the task (in a synchronous environment), or the total distance moved by the robots to finish it (in an asynchronous setting). Patrolling has a different "cost", the time between consecutive visits to any point in the region, the so-called "idle time".

A little explored feature of the robots is their *speed*. Most past work has focused on the case where all robots share the same (maximal) speed. Notable exceptions of which the authors are aware include [11], which considers the evacuation problem on the infinite line with robots with distinct maximal speeds, [14], which introduces a non-intuitive ring patrolling strategy using three robots with distinct maximal speeds, and [20, 25], where the rendezvous problem with different speeds in a cycle is studied. It is this feature, robots with different maximal speeds, that we explore in this paper. Such a feature makes our model more general and applicable to real-life scenarios, e.g., .

The most relevant line of work explores the evacuation problem in the unit disk with two robots with identical speeds. The wireless communication model is studied in [13], where they provide an optimal evacuation strategy. The face-to-face communication model is examined in [13, 18, 9]. In this case, the strategies provided are nearly optimal, yet exact optimality seems to be very difficult to obtain. Hence, a more recent work [12] turns the attention to average-case analysis and discusses its trade-offs with respect to worst-case.

Recently, many variations of the problem have appeared in the literature such as first locating a treasure and then evacuating it via the exit [22, 23], evacuating a designated robot first due to security priorities [16, 17], evacuating via two unknown exits [31], and evacuating in the presence of a faulty robot [15].

1.2 Our results

We consider the evacuation problem in the unit disk using two robots with distinct maximal speeds: one with speed 1, the second with speed $s \geq 1$. The robots share a common clock and can communicate instantaneously when they have found the exit (wireless communication) and so can synchronize their behavior in the evacuation procedure. We assume that the robots can measure distances to an arbitrary precision (equivalently, they can measure time to an arbitrary precision), and can vary their speeds as they desire, up to their maximum speed. A necessity for robots to travel with less than optimal speed could emerge if further constraints are added to the model, e.g., communication radius restrictions where a faster robot might need to slow down to remain near a slower robot in order to be able to maintain an open communication channel. Note that, in our bounds to follow, the robots always travel at maximum speed.

We show that, even in the case of two robots, the analysis involved in finding (time) optimal evacuation strategies can become intricate with strategies that depend on the ratio of the fast robot's to the slow robot's maximal speed. For large s, we introduce an efficient and non-obvious search strategy, called the *Half-Chord* Strategy, see Figure 1. For small s, we generalize a strategy from [13], namely the *Both-to-the-Same-Point* Strategy (BSP), where the two robots move to the same point on the boundary and then separately explore the boundary in clockwise and counterclockwise directions to find the exit (Figure 5a). For values of $s \geq c_{1.86}$ (with $c_{1.86} \approx 1.856$), we show that BSP is not optimal by demonstrating that the Half-Chord Strategy is superior to it. Moreover, we improve on this with the *Fast-Chord* Strategy (Figure 7), which outperforms Half-Chord for $1.71 \approx c_{1.71} < s < c_{2.07} \approx 2.07$. We obtain optimality for all $s \geq c_{2.75} \approx 2.75$, in the wireless setting, as we demonstrate matching upper and lower bounds on the evacuation time. For $s \in (1, c_{2.75})$, we provide lower bounds on the evacuation time that do not match the upper bounds provided by the respective search strategies (BSP for $s < c_{1.71}$, Fast-Chord for $s \in [c_{1.71}, c_{2.07})$, and Half-Chord for $s \geq c_{2.07}$). The worst ratio between our upper and lower bound, 1.22, is realized for $s = c_{1.71}$.

Section 2 contains a formal definition of the problem we consider. Section 3 contains our upper bounds on the evacuation time, while Section 4 has our lower bounds. Finally, Section 5 concludes with remarks about optimality, by comparing our prevailing upper and lower bounds, as well as some further work suggestions.

2 Problem Definition and Strategy Space

In this section, we detail the proposed environment/model, and define the optimization problem in question. Also, we provide a partition of the strategy space and some other useful observations.

The continuous environment in which all the action takes place is the unit disk, i.e., a disk of radius 1. Two robots, called *Fast* and *Slow*, are initially placed at the center of the disk. Fast has some maximal speed $s \geq 1$ ($s \in \mathbb{R}$), whereas Slow has maximal speed 1. A robot is aware of its identity being Fast or Slow, and so of its maximal speed. The robots occupy the size of a single point, are allowed to move on and within the boundary of the disk at a speed up to their maximal one, and can communicate wirelessly, and instantaneously, at any time. Moreover, the robots can recognize whether their current location is an interior or a boundary point, can measure distances and time to an arbitrary precision, and have a common perception of time. The goal of the robots is to discover the *exit*, i.e., an unknown to them boundary point. Either robot can instantaneously perceive whether a boundary point is the exit, or not, only when placed exactly on it. The *evacuation time* is the time it takes, after starting from the center, till both robots have reached the exit. From now on, we refer to the model described thus far as the *fast evacuation model* and all discussion that follows is with respect to this model.

A *main evacuation strategy* is a pair of algorithms, one per robot, which describe how each robot moves until the point where one of the robots has discovered the exit. A main evacuation strategy is simply a description of the movement of both robots that respects their speed limits, such that every point on the boundary is visited by at least one robot (unless the exit is discovered before the whole boundary needs to be explored). Therefore, the exit is surely discovered, meaning that a main evacuation strategy is a *feasible* strategy in this respect, i.e., both robots will evacuate the disk within finite time. A *full evacuation strategy* is a pair of algorithms, one per robot, which describe how each robot moves such that both robots reach the exit, and so evacuate the disk, by the end of execution. A full evacuation strategy includes, for each possible exit point, a description of the movement of the robots after the discovery of the exit point. However, as a consequence of the disk environment and wireless communication, optimal movement in this second part for the not-yet-on-the-exit robot is to traverse a straight line to the exit, cf. Remark 1. Hence, it suffices to focus on designing a time-efficient main evacuation strategy.

Remark 1. *In any full evacuation strategy, when one robot discovers the exit, from that moment on, the optimal trajectory to be executed by the other robot is to follow a straight line to the exit.*

To formally define the optimization problem below, let the *worst-case evacuation time* for a main/full evacuation strategy be the *maximum* evacuation time taken over all possible exit positions.

Definition 1 (Fast Evacuation Problem)**.** *Given a real number $s \geq 1$, design a main evacuation strategy such that the worst-case evacuation time (given as a function of s) in the fast evacuation model is minimized.*

We now proceed with identifying key aspects of potential (main) evacuation strategies.

Definition 2. *In a "fast-explores" evacuation strategy, for all exit positions, Slow does not reach the disk boundary before the exit is discovered by Fast. We define the set of all fast-explores strategies as FES.*

Definition 3. *In a "slow-explores" evacuation strategy, for all exit positions, Fast does not reach the disk boundary before the exit is discovered by Slow. We define the set of all fast-explores strategies as SES.*

Definition 4. *In a "both-explore" evacuation strategy, there exists some exit position such that Slow and Fast visit the boundary before the exit is discovered. We define the set of all both-explore strategies as BES.*

Notice that, for $s = 1$, if only one robot explores the boundary, we randomly assign such a strategy to FES or SES. Below, let ALL stand for the set of all evacuation strategies.

Proposition 1. *(BES, FES, SES) forms a partition of ALL.*

Proof. BES, FES and SES are pairwise disjoint, since if and only if either Fast alone or Slow alone visit the boundary before the exit is discovered, then both do not, and if and only if Fast visits the boundary before the exit is discovered, then Slow does not.

$ALL = BES \cup FES \cup SES$, since for any possible evacuation strategy either Fast (for all exit positions) or Slow (for all exit positions) or both (for some exit positions) visit the boundary before the exit is discovered. □

Note that an optimal SES strategy is obviously to have Slow reach the boundary and explore it (counter) clockwise. Fast follows Slow up to an infinitesimally small distance $\epsilon > 0$ always staying within the disk interior. The worst-case evacuation time is $1 + 2\pi + \epsilon$ and it occurs when the exit is just missed by Slow. In the sections to follow, we utilize the speed advantage of Fast in order to prove much better bounds.

Notation and Terminology. In the sections to follow, we denote a line segment from point A to point B as AB and its length as $|AB|$. Disk (O, r) denotes a disk centered at O with radius r. A (counter clockwise) circular arc from A to B is denoted $\widehat{AB}$ with length $|\widehat{AB}|$. $\measuredangle AOB$ denotes the (counter clockwise) central angle formed by line segments AO and BO at disk center O: starting from AO and moving counter clockwise toward BO. Whenever we consider an arc or angle in clockwise fashion instead, we expressly cite so in the text. For three endpoints X, Y, Z, we use the notation $\triangle XYZ$ to denote a triangle and we discuss in-triangle angles, unless otherwise stated. In the strategy figures, double arrows (in blue) indicate trajectories followed by Fast, whereas single arrows (in red) indicate trajectories followed by Slow.

3 Upper Bounds

As a warm-up, consider a "mimic" strategy as a first fast-explores strategy. Both robots set out from the disk center toward the boundary on the same direction each with maximum speed. When Fast reaches the boundary, Slow stops as well. From now on, as Fast explores the boundary in (counter) clockwise fashion, Slow "mimics" Fast's movement by moving on the boundary of a smaller disk with the same center, but with radius $1/s$ instead of 1. When Fast discovers the exit, Slow lies on the corresponding point on the smaller disk and takes a direct line segment of length $1 - 1/s = (s - 1)/s$ to it, i.e., Slow moves for the remaining part of a unit length radius. Overall, in the worst case, Fast just misses the exit and has to traverse the whole boundary. It takes $1/s$ time for Fast to initially reach the boundary, $2\pi/s$ for Fast to traverse the boundary and another $(s - 1)/s$ for Slow to reach the exit. Altogether, we get a $1 + 2\pi/s$ evacuation time.

In the next subsection, we present a more convoluted strategy which outperforms the just described one. Intuitively, the improvement is derived by modifying the behavior of Slow. Instead of mimicking Fast throughout the whole boundary exploration, Slow now tries to be near enough to Fast only during the final stages of exploration, that is, when the worst case scenario emerges.

3.1 The Half-Chord Strategy for $s \in [2, \infty)$

We now present an FES strategy which we later prove optimal for big enough values of s. The idea for this strategy stems from the proof of the FES lower bound to follow (Theorem 4). When there exists a long chord between two yet unexplored endpoints, an adaptive adversary might place the exit in either endpoint for Fast to discover. In this case, the optimal play for Slow is to be on the midpoint of this chord in order to minimize the time needed till it also reaches the exit.

In the *Half-Chord* strategy, Fast reaches the boundary and explores it counter clockwise. On the other hand, Slow follows a trajectory with nice properties and reaches the midpoint of a carefully chosen chord whose endpoints capture worst case evacuation scenarios. Slow reaches the chord midpoint exactly when Fast reaches one of the endpoints (Proposition 2). Then, in the worst-case, Slow has to traverse half the length of this chord to reach the exit (after the exit location is communicated by Fast).

The worst-case analysis is performed for $s \in [2, \infty)$. For the strategy details below, please refer to Figure 1. The center of the disk is denoted by O. Fast's trajectory is given with *double arrows*, while Slow's with *single arrows*. Unless otherwise stated, all angles and arcs are considered in counterclockwise order.

3.1.1 The Strategy

Initially, both robots lie at the center of the disk at time $t = 0$. They then move in straight lines with an angle of $\phi := \pi + 1/2$ between them until Fast reaches the boundary, that is, for $\frac{1}{s}$ time.

Let B be the first boundary point reached by Fast, that is, at time $t = \frac{1}{s}$. From now on, Fast's strategy is to explore the boundary in counter clockwise fashion, until it reaches point B again at time $t = \frac{1+2\pi}{s}$.

On the other hand, Slow continues on its straight line for another $\frac{1}{s}$ time until it reaches point C (Phase I), where $|OC| = \frac{2}{s}$. Note that $\phi = \measuredangle BOC$, since Slow does not divert from its initial straight-line trajectory.

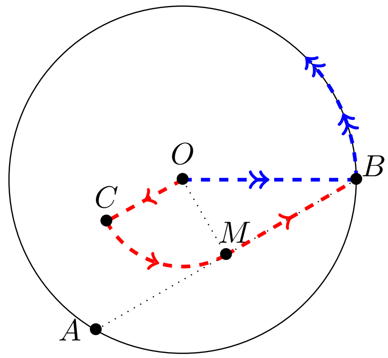


Figure 1: The Half-Chord Strategy (example depiction for $s = 4$)

Afterward, for another $\frac{2\arccos(-2/s)-1}{s}$ time, Slow takes an arc from C to M on the disk with radius $\frac{2}{s}$ centered at O; from now on referred to as disk $(O, \frac{2}{s})$ (Phase II). Finally, Slow traverses MB until time $t = \frac{1+2\pi}{s}$ (Phase III) when Fast re-reaches B, i.e., the whole boundary has been explored. Note that, in Figure 1, A is the point with arc distance $2\arccos\left(-\frac{2}{s}\right)$ from B.

In Algorithm 1, respectively Algorithm 2, we provide a more structured and formal main evacuation strategy for Fast, respectively Slow. Bear in mind that if at any time Fast locates the exit, it instantly terminates and informs Slow of the exit location. Then, due to Remark 1, Slow's strategy reduces to following a straight line to the exit.

Algorithm 1: Half-Chord for Fast robot

1: Traverse line segment OB // $t = 0 \ldots 1/s$
2: Traverse disk boundary counterclockwise // $t = 1/s \ldots (1+2\pi)/s$

Algorithm 2: Half-Chord for Slow robot

1: Traverse line segment OC // Phase I : $t = 0 \ldots 2/s$
2: Traverse arc $\widehat{CM}$ on disk $\left(O, \frac{2}{s}\right)$ // Phase II : $t = 2/s \ldots (1 + 2\arccos(-2/s))/s$
3: Traverse line segment MB // Phase III: $t = (1 + 2\arccos(-2/s))/s \ldots (1 + 2\pi)/s$

Proposition 2. *Fast reaches A exactly when Slow reaches M.*

Proof. Fast reaches A after $\frac{1+2\arccos(-2/s)}{s}$ time, since it takes $\frac{1}{s}$ time for it to traverse OB and $\frac{2\arccos(-2/s)}{s}$ time to traverse $\widehat{BA}$. Slow reaches C after time $\frac{2}{s}$. Then, by Algorithm 2, it traverses $\widehat{CM}$ for another $\frac{1}{s}(2\arccos(-2/s) - 1)$ time for a total of $\frac{1+2\arccos(-2/s)}{s}$. □

Proposition 3. *M is the midpoint of chord AB.*

Proof. By the strategy, we get $|\widehat{CM}| = \frac{1}{s}(2\arccos(-2/s) - 1)$. Since we work on disk $(O, \frac{2}{s})$, the corresponding angle is $\measuredangle COM = \frac{s}{2}|\widehat{CM}| = \arccos(-2/s) - 1/2$. Let us now consider a parametric representation of the two disks $(O, 1)$ and $(O, \frac{2}{s})$. In such a representation, based on our strategy, we get the following coordinates for point C:

$$C = \left(\frac{2}{s}\cos(\pi + 1/2),\ \frac{2}{s}\sin(\pi + 1/2)\right)$$

Given our knowledge of $\measuredangle COM$, we can extract the coordinates for M as:

$$\begin{aligned} M &= \left(\tfrac{2}{s}\cos(\pi+1/2+\arccos(-2/s)-1/2),\ \tfrac{2}{s}\sin(\pi+1/2+\arccos(-2/s)-1/2)\right) \\ &= \left(-\tfrac{2}{s}\cos(\arccos(-2/s)),\ -\tfrac{2}{s}\sin(\arccos(-2/s))\right) \\ &= \left(\tfrac{4}{s^2},\ -\tfrac{2}{s}\sqrt{1-\tfrac{4}{s^2}}\right) \end{aligned}$$

Now, let us consider points A, B. By the parametric representation, we get the coordinates:

$$A = (x_A, y_A) = (\cos(2\arccos(-2/s)), \sin(2\arccos(-2/s))) = \left(\tfrac{8}{s^2}-1, -\tfrac{4}{s}\sqrt{1-\tfrac{4}{s^2}}\right)$$

$$B = (x_B, y_B) = (\cos(0), \sin(0)) = (1, 0)$$

Let us now consider the midpoint of chord AB, namely some point $M' = (x_{M'}, y_{M'})$. We get $x_{M'} = (x_A + x_B)/2 = (8/s^2 - 1 + 1)/2 = 4/s^2$ and $y_{M'} = (y_A + y_B)/2 = (-\frac{4}{s}\sqrt{1-\frac{4}{s^2}} + 0)/2 = -\frac{2}{s}\sqrt{1-\frac{4}{s^2}}$. Noticing that $x_M = x_{M'}$ and $y_M = y_{M'}$ completes the proof. □

Proposition 4. *Fast explores the whole boundary before Slow reaches B.*

Proof. Slow reaches M after $\frac{1+2\arccos(-2/s)}{s}$ time and then has to traverse the line segment MB. By Proposition 3, $|MB| = |BA|/2 = 2\sin(\widehat{BA}/2)/2 = \sin(2\arccos(-2/s)/2) = \sqrt{1-\frac{4}{s^2}}$. Meanwhile, at time $\frac{1+2\arccos(-2/s)}{s}$, Fast lies on A and then has to traverse $\widehat{AB}$ for another $\frac{2\pi-2\arccos(-2/s)}{s}$. It's adequate to see that $\sqrt{1-\frac{4}{s^2}} \geq \frac{2\pi-2\arccos(-2/s)}{s}$ for any $s \geq 2$. Consider $f(s) = \sqrt{1-\frac{4}{s^2}} - \frac{2\pi-2\arccos(-2/s)}{s} = \sqrt{1-\frac{4}{s^2}} - \frac{2\arccos(2/s)}{s}$. It suffices to notice that $f(2) = 0$ and that $\frac{df}{ds} = \frac{2\arccos(2/s)}{s^2} \geq 0$ for any $s \geq 2$. □

The aforementioned proposition, together with the fact that it takes $\frac{1+2\pi}{s}$ time for Fast to explore the whole boundary, provides us with the endtime for Phase III and the strategy in general.

The main result of this section below follows from the combination of the upper bounds later proved for Phases I (Lemma 1), II (Lemma 2), and III (Lemma 3).

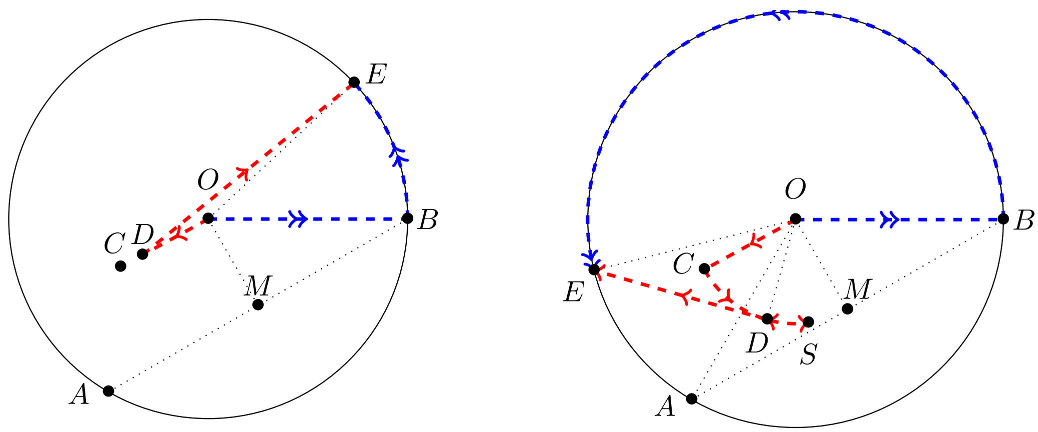
Theorem 1. *For any $s \geq 2$, the worst-case evacuation time of the Half-Chord strategy is at most*

$$\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$$

3.1.2 Phase I

Lemma 1. *The Half-Chord evacuation strategy takes at most $\frac{(1+2\arccos(-2/s))}{s} + \sqrt{1-\frac{4}{s^2}}$ evacuation time, if the exit is found during Phase I.*

Proof. We need only care about the time $t \in [1/s, 2/s]$, since for less time Fast has not yet reached the boundary. Imagine that the exit is discovered after $(1+a)/s$ time (for $a \in [0,1]$). For a visualization, the reader can refer to Figure 2a. Slow has traversed $(1+a)/s$ distance on the OC segment, while Fast has explored an a part of $\widehat{BA}$. Slow now takes a segment from its current position (namely D) to the exit E. To compute $|DE|$ we use the law of cosines in $\triangle DOE$. Let $\omega = \measuredangle DOE$. Also, to help us with the proof, let $\theta := \measuredangle COM = \frac{s}{2}|\widehat{CM}| = \arccos\left(-\frac{2}{s}\right) - 1/2$ and $\psi := \measuredangle MOB = 2\pi - \phi - \theta = \pi - \arccos\left(-\frac{2}{s}\right)$. We distinguish two cases based on the value of a. In case $a \leq \frac{1}{2}$, then $\omega \leq \pi$, and more accurately $\omega = a + \psi + \theta = \pi + a - \frac{1}{2}$.



(a) Exit during Phase I ($a = 0.75$)

(b) Exit during Phase II ($\tau = 0.2$)

Figure 2: Exit during Phases I & II (Examples for $s = 4$)

In case $a > \frac{1}{2}$, then $\omega > \pi$, and more accurately $\omega = 2\pi - a - \psi - \theta$. Since $\cos(2\pi - x) = \cos(x)$, we can consider the two cases together in the distance calculation below. We compute,

$$|DE| = \sqrt{|OE|^2 + |OD|^2 - 2|OE||OD|\cos(\omega)} = \sqrt{1 + \frac{(1+a)^2}{s^2} + 2\frac{1+a}{s}\cos(1/2 - a)}$$

Overall, the worst-case evacuation time is given by

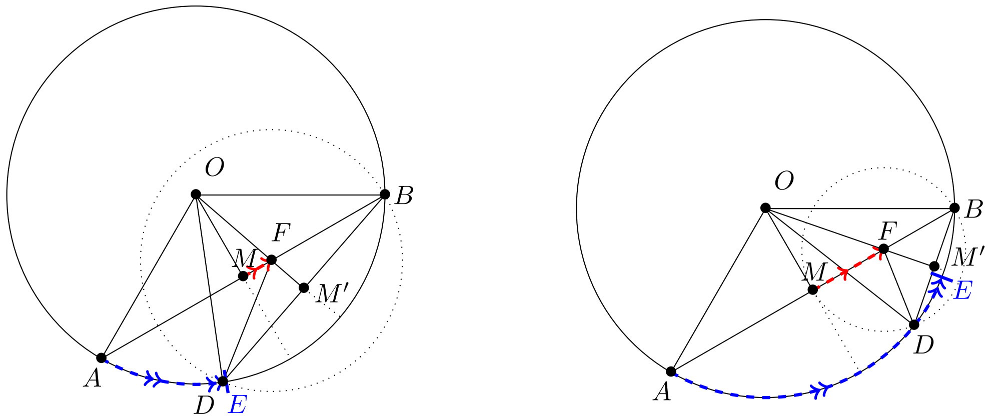
$$\max_{a \in [0,1]} \left\{ \frac{1+a}{s} + \sqrt{1 + \frac{(1+a)^2}{s^2} + 2\frac{1+a}{s}\cos(1/2 - a)} \right\}.$$

To conclude the proof, it suffices to observe that $\frac{2}{s} + \sqrt{1 + \frac{2^2}{s^2} + 2\frac{2}{s}} = \frac{s+4}{s}$ is an upper bound to the above quantity, since $a \leq 1$ and $\cos(\cdot) \leq 1$. Finally, one can verify $\frac{s+4}{s} \leq \frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1 - \frac{4}{s^2}}$ for any $s \geq 2$. Let $f(s) = \frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1 - \frac{4}{s^2}} - \frac{s+4}{s} = \frac{2\arccos\left(-\frac{2}{s}\right) + \sqrt{s^2-4} - s - 3}{s}$. In the interval $[2, \infty)$, we get $f(2) = \pi - \frac{5}{2} > 0$, $\frac{d}{ds} f(s) = \frac{3 - 2\arccos(-2/s)}{s^2} < 0$, for all $s \geq 2$, and $\lim_{s \to \infty} f(s) = 0$. $\square$

3.1.3 Phase II

Lemma 2. *The Half-Chord evacuation strategy takes at most* $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1 - \frac{4}{s^2}}$ *evacuation time, if the exit is found during Phase II.*

Proof. We prove that the worst-case placement for the exit is point A. For the details below, refer to Figure 2b. Suppose the exit E is found at the time when Slow lies on point S and a τ part of $\widehat{CM}$ remains to be traversed. The corresponding central angle is $\frac{s\tau}{2}$, since $\widehat{CM}$ is an arc on $(O, \frac{2}{s})$. At the same time, Fast has not yet searched an $s\tau$ part of $\widehat{BA}$ with a corresponding central angle of size $s\tau$. For the sake of the analysis, we describe a Slow trajectory that does *not* go directly to the exit when it is announced, rather Slow follows a more costly trajectory which turns out to be good enough for our bound. That is, Slow moves backwards on the boundary of $(O, \frac{2}{s})$ for another τ distance to point D and then traverses $|DE|$. Note that the central angle from D to M is $\frac{s\tau}{2} + \frac{s\tau}{2} = s\tau$ and matches the central angle between E and A. Thence, due to shifting by the same central angle, we get $\measuredangle EOD = \measuredangle EOA + \measuredangle AOD = \measuredangle DOM + \measuredangle AOD = \measuredangle AOM$. Moreover, since $|OD| = |OM| = \frac{2}{s}$ and $|OE| = |OA| = 1$, triangles $\triangle EOD$ and $\triangle AOM$ are congruent meaning that $|ED| = |AM|$. To sum up, if the exit is discovered τ time before Slow reaches M, from then on, it takes at most $\tau + \sqrt{1 - \frac{4}{s^2}}$ time for it to reach it. At the same time, it would take $\tau + \sqrt{1 - \frac{4}{s^2}}$ time for it to reach A. Hence, exiting through A is the worst-case scenario and yields a total time of $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1 - \frac{4}{s^2}}$. $\square$



(a) First case (example for $\tau = \frac{1}{5}$)

(b) Second case (example for $\tau = \frac{1}{2}$)

Figure 3: Exit during Phase III (example for $s = 4$; exit E lies at the end of Fast's arrow)

3.1.4 Phase III

Lemma 3. *The Half-Chord evacuation strategy takes at most* $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$ *evacuation time, if the exit is found during Phase III.*

Proof. Since $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s}$ time has passed at the beginning of Phase III, it suffices to show that at most another $\sqrt{1-\frac{4}{s^2}}$ time goes by till Slow reaches the exit, when the exit is discovered within $\widehat{AB}$.

Suppose that the exit is discovered τ time units after the beginning of Phase III. Then, Slow lies at F (Figure 3), τ distance away from M on the MB segment. On the other hand, Fast lies on E, an $s\tau$ distance away from A on $\widehat{AB}$ (at the end of the double arrow arc in Figure 3).

We demonstrate that the worst case scenario is that the exit was "just missed" by Fast when it reached point B, i.e., the exit lies on a boundary point infinitesimally near to the clockwise of point B. Consider a disk with center F and radius $r = \sqrt{1-\frac{4}{s^2}} - \tau$, that is, r captures the remaining distance $|FB|$ for Slow to reach B. One can notice that (F, r) intersects $(O, 1)$ at two points: one of them is B and the other one is D, where D is included in $\widehat{AB}$, since $|AF| \geq r$ for any choice of $\tau \geq 0$. Moreover, we draw the chord DB and its middle point, say M'. Now, notice that OM' is perpendicular to DB, since DB is a chord of $(O, 1)$ and also that OM' passes through F, since DB is also a chord of (F, r). To conclude, we exhibit that E is included in $\widehat{DB}$. Equivalently, that $|\widehat{AE}| \geq |\widehat{AD}|$. We look into two cases.

First, that $\measuredangle AOD \leq \measuredangle AOM$. In this case, we compute

$$
\begin{aligned}
\measuredangle AOD &= \measuredangle AOM - \measuredangle DOM \\
&= \measuredangle MOB - \measuredangle DOM \\
&= \measuredangle MOM' + \measuredangle M'OB - \measuredangle DOM \\
&= \measuredangle MOM' + \measuredangle DOM' - \measuredangle DOM \\
&= 2 \cdot \measuredangle MOM'
\end{aligned}
$$

since $\measuredangle AOM = \measuredangle MOB$ and $\measuredangle M'OB = \measuredangle DOM'$ from the fact that OM (OM') bisects AB (DB). Moreover, $\measuredangle DOM' - \measuredangle DOM = \measuredangle MOM'$. We compute $\measuredangle MOM' = \arctan(s\tau/2)$ by the right triangle $\triangle MOF$. Finally, $\measuredangle AOD = 2\arctan(s\tau/2) \leq s\tau = \measuredangle AOE$, since $\arctan(x) \leq x$ for $x \geq 0$.

For the second case, let $\measuredangle AOD > \measuredangle AOM$. Then, $\measuredangle AOD = \measuredangle AOM + \measuredangle MOD = \measuredangle MOB + \measuredangle MOD = \measuredangle MOM' + \measuredangle M'OB + \measuredangle MOD = \measuredangle MOM' + \measuredangle DOM' + \measuredangle MOD = 2 \cdot \measuredangle MOM'$, again by using the equalities deriving from bisecting the chords. The rest of the proof follows exactly as before. $\square$

3.2 The Half-Chord Strategy for $1 \le s \le 2$

To have a complete picture, let us consider a *generalized* Half-Chord strategy which works for small values of s. We first observe that, for $s = 2$, the name "Half-Chord" is slightly misleading, as the points A, B, and M, coincide. Recall that A is the point with arc distance $2\arccos\left(-\frac{2}{s}\right)$ from B. Then, for $s = 2$, it holds $2\arccos\left(-\frac{2}{2}\right) = 2\pi$, i.e., A and B coincide. Hence, Slow's strategy becomes to simply traverse $\widehat{CB}$; see Figure 4. In this respect, the Half-Chord strategy for $s = 2$ is a BES strategy with a worst case evacuation time of at most $\frac{1+2\pi}{2}$, which results by substituting $s = 2$ in the bound in Theorem 1.

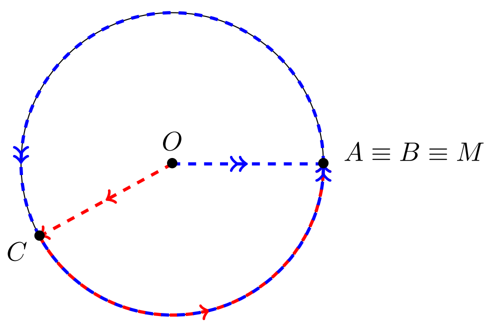


Figure 4: The (generalized) Half-Chord strategy for $s \in [1, 2]$

In case $s = 2$, Fast travels at a speed $s/2$ times faster than Slow on their defined trajectories. For $s < 2$, we can stick to the exact same trajectories for the robots, by enforcing Slow to move even slower than its maximum speed, namely at a speed $\frac{s}{2} < 1$ such that the $s/2$ ratio of the two speeds is maintained. In case $s = 2$, for any time t, consider the location of Slow at time t, say x, and the location of Fast at time t, say y. Then, in case $s < 2$, since the exact same trajectories are followed by Fast and Slow but at a $s/2$-factor lesser speeds, Slow lies on x and Fast lies on y at time $t \cdot 2/s$. Therefore, the set of all pairs (x, y) of current positions for Fast and Slow that appear at any time during the execution of the strategy remains the same. Hence, the worst case exit point is preserved and the corresponding evacuation time will be $2/s$ times greater than the worst case evacuation time for $s = 2$, i.e., $2/s \cdot \frac{1+2\pi}{2}$. By these observations, Corollary 1 follows.

Corollary 1. *For any real s, where $1 \le s \le 2$, the worst-case evacuation time for the (generalized) Half-Chord strategy is at most $\frac{1+2\pi}{s}$.*

3.3 The Both-to-the-Same-Point Strategy

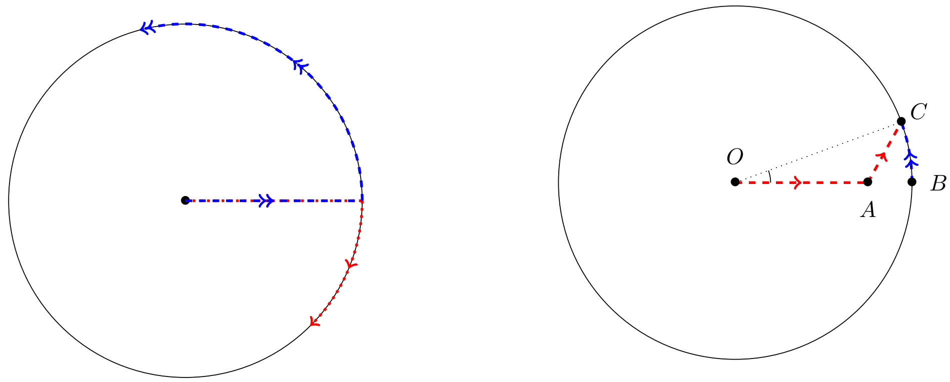
In this subsection, we propose a BES strategy, which follows the same key idea as the strategy presented in [13], where it was proven optimal for the case $s = 1$, i.e., the equal-speeds case.

3.3.1 The Strategy

In the *Both-to-the-Same-Point* strategy, shortly *BSP*, initially both robots set out toward the same boundary point, moving in a straight line. Once they arrive there, they move in opposite directions along the boundary. Without loss of generality, Fast moves counterclockwise along the boundary, while Slow moves clockwise. This goes on, until the exit has been found by either robot or the robots meet each other on the boundary, and so the whole boundary has been searched. For a visualization of the strategy, see Figure 5a. Fast's trajectory is given in blue (double arrows), while Slow's in red (single arrows). Below, we restrict the analysis of BSP for $s \in [1, 2]$, since for $s > c_{1.71} \approx 1.71$ we later show that this strategy is outperformed.

3.3.2 Exit Before Slow Explores

Lemma 4. *It takes at most $1 + \sqrt{2 - 2\cos(s-1)}$ time (where $s \in [1, 2]$) for both robots to evacuate in the BSP strategy, when the exit is found before Slow has reached the boundary.*



(a) The BSP Strategy

(b) Exit before Slow explores
(Example for $s = 1.8$ and $a = 0.35$)

Figure 5: The BSP Strategy and an Evacuation Example

Proof. Let a stand for the distance Fast has explored on the boundary before finding the exit. Notice that $a \le s - 1 \le 1$, since a stands for some already searched distance before Slow reaches the boundary. The total evacuation time is the time needed for Fast to find the exit and then for Slow to reach it. Let b stand for the latter. Then, the worst-case evacuation time is $\max_{0 \le a \le s-1} \left\{ \frac{a+1}{s} + b \right\}$, where $b = \sqrt{1 + \left(\frac{a+1}{s}\right)^2 - 2 \cdot \frac{a+1}{s} \cos(a)}$ by the cosine law in the formed triangle ($\triangle OAC$ in Figure 5b with $|OC| = 1$, $|OA| = \frac{1+a}{s}$ and $\measuredangle AOC = a$). The maximum is attained for $a = s - 1$ and, for this value of a, we compute $\frac{a+1}{s} + b = 1 + \sqrt{2 - 2\cos(s-1)}$ (see appendix). $\square$

3.3.3 Exit After Slow Explores

Lemma 5. *In the BSP strategy (where $s \in [1, 2]$), consider the time t when the exit is found, after Slow has explored some part of the boundary. Then, the evacuation time is at most*

- $\frac{2s+\pi+4}{s+1}$, *when the angle between the two robots is less or equal to π at time t and*
- $1 + 2\sqrt{1 - \frac{1}{(s+1)^2}} + \frac{2\arccos(\frac{1}{-s-1}) - s + 1}{s+1}$ *when the angle is between π and 2π at time t.*

Proof. Suppose some time $1 + d$ has passed since the beginning of execution, for some $d \ge 0$. At this time, Slow has reached the boundary (at time 1) and has searched it clockwise for another d time at speed 1. Meanwhile, Fast has searched a $s - 1 + sd$ distance on the boundary, since it reaches the boundary at time $1/s$ and then searches the boundary for another $1 + d - 1/s$ time at speed s, that is, searching a distance of $s(1 + d - 1/s) = s - 1 + sd$. Put together, an $s - 1 + d + sd$ distance on the boundary has been searched thus far. Let $angle(d, s) := s - 1 + d + sd$, since the quantity also represents the angle between Fast and Slow from the center of disk. Now, suppose that the exit is discovered by Fast at time $1 + d$. We break the analysis into two cases:

- $angle(d, s) \le \pi$:
  In this case, $s - 1 + d(s+1) \le \pi$, which results to $d \le \frac{\pi - s + 1}{s+1}$. Notice that the bound of d is non-negative for $s \le 2$. The worst-case evacuation time is given by computing the function

$$\max_{0 \le d \le \frac{\pi - s + 1}{s+1}} \left\{ 1 + d + 2\sin\left(\frac{d(s+1) + s - 1}{2}\right) \right\}$$

  where the last addend accounts for the chord length needed to be searched by Slow. The maximum is attained at $d = \frac{\pi - s + 1}{s+1}$ for a worst-case time of $\frac{2s+\pi+4}{s+1}$ (see appendix).

- $\pi < angle(d,s) < 2\pi$:
  In this case, $d \in (d_{min}, d_{max}) = (\frac{\pi-s+1}{s+1}, \frac{2\pi-s+1}{s+1})$. The function to be maximized is the same as above. The maximum is attained at $d' = \frac{2\cdot\arccos(-1/(s+1))-s+1}{s+1}$ yielding an evacuation time $1+2\sqrt{1-\frac{1}{(s+1)^2}}+\frac{2\arccos(\frac{1}{-s-1})-s+1}{s+1}$ (see appendix).

Finally, we need not care about the case where Slow finds the exit, since the time taken for Fast to traverse the same chord will be less than the worst-case scenario examined. □

3.3.4 Comparison

For any $s \in [1,2]$, the maximum (worst-case) upper bound comes from the second case of Lemma 5 and yields the result in Theorem 2; see Figure 6.

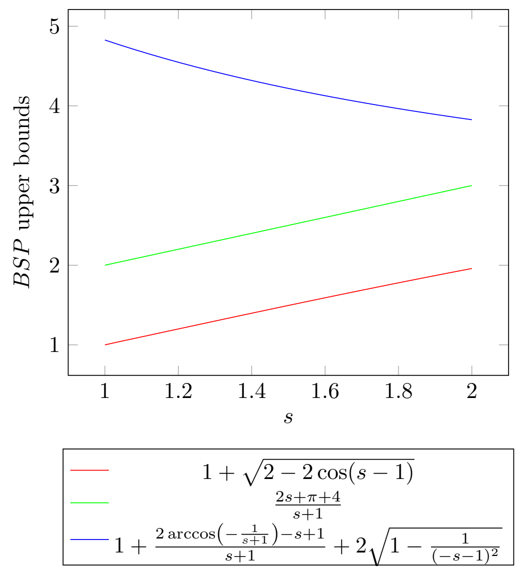


Figure 6: Comparison of BSP upper bounds derived in Lemmata 4 and 5 for $s \in [1,2]$

Theorem 2. *For any $s \in [1,2]$, BSP requires evacuation time at most*

$$1+2\sqrt{1-\frac{1}{(s+1)^2}}+\frac{2\arccos(-\frac{1}{s+1})-s+1}{s+1}.$$

3.4 The Fast-Chord Strategy for $s \leq 2\pi+1$

Recall that in Half-Chord for $s=2$, we observe that the final point reached after Phase I, i.e., point C, lies on the disk boundary. Thence, after that, Slow explores $\widehat{CB}$, but so does Fast, since by its strategy it explores the whole boundary. This seems to be an unnecessary double exploration of this part of the boundary. Thus, we propose a new strategy, where Fast reaches C as usual, but then traverses the CB chord, instead of $\widehat{CB}$. Furthermore, we could vary the position of C, in order for Fast to reach B (for the second time) exactly when Slow reaches D (a point before B) and so get Fast to explore some part of the boundary in clockwise fashion as well. In this case, Slow does not traverse the whole $\widehat{CB}$. Let us now describe more formally this *Fast-Chord* family of strategies. All arcs are considered in *counterclockwise* fashion unless otherwise stated. Below, let $|\widehat{BA}|= s-1$, $x_1 = |\widehat{AC}|$, $x_2 = |CB|$, $x_3 = |\widehat{DB}|$ and $y = |\widehat{CD}|$; see Figure 7. In Algorithms 3, 4, we define the main evacuation strategy followed by Fast and Slow for Fast-Chord.

The following system of equations describes the relationship between the variable distances:

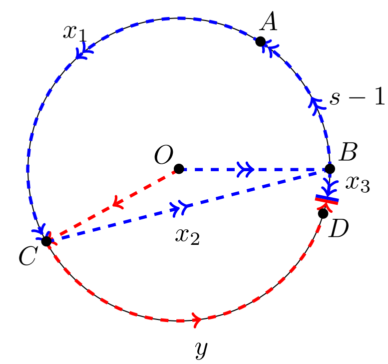


Figure 7: The Fast-Chord Family of Strategies

Algorithm 3: Fast-Chord for Fast robot

1: Traverse line segment OB	// $t = 0 \dots 1/s$
2: Traverse arc $\widehat{BA}$	// Phase I: $t = 1/s \dots 1$
3: Traverse arc $\widehat{AC}$	// Phase IIa: $t = 1 \dots 1 + x_1/s$
4: Traverse line segment CB	// Phase IIb: $t = 1 + x_1/s \dots 1 + (x_1 + x_2)/s$
5: Traverse (clockwise) arc $\widehat{BD}$ till you meet Slow	// Phase IIc: $t = 1 + (x_1 + x_2)/s \dots 1 + (x_1 + x_2)/s + x_3/(s+1)$

$$\begin{cases} x_1 + y + x_3 + s - 1 & = 2\pi & \text{(I)} \\ x_2 & = 2\sin\left(\frac{x_3+y}{2}\right) & \text{(II)} \\ x_1 + x_2 & = s \cdot y & \text{(III)} \end{cases}$$

Equation (I) suggests how the disk boundary is partitioned. Equation (II) suggests that x_2 is the chord of an arc with length $x_3 + y$. Equation (III) suggests that Fast traverses x_1 and x_2 at the same time as slow traverses y. That is, since Fast lies on A exactly when Slow lies on C, then Fast arrives at B (for the second time) exactly when Slow arrives at D. The latter happens at time $1 + y = 1 + \frac{x_1+x_2}{s}$. The remaining x_3 part of the boundary can be explored in time $\frac{x_3}{s+1}$, since both robots explore it concurrently until they meet. Hence, within $\frac{x_3}{s+1}$ time, they can explore a distance equal to $s \cdot \frac{x_3}{s+1} + \frac{x_3}{s+1} = (s+1) \cdot \frac{x_3}{s+1} = x_3$. All variables are non-negative representing distance. Note that the above system of equations is only valid for our purposes here for $s < 2\pi + 1$. In case $s \geq 2\pi + 1$, then Fast searches the whole boundary and discovers the exit before Slow reaches the boundary, therefore it would hold $|\widehat{BA}| = s - 1 \geq 2\pi$.

The idea behind this paradigm is to try different values for x_3 and then solve the above system to extract x_1, x_2 and y. Nonetheless, due to the $\sin(\cdot)$ function in equation (II), we could not obtain a symbolic solution. Thence, we hereby provide bounds computed *numerically*[1]. For any value of s, we iterate over all possible x_3 values (using some discrete step/accuracy, e.g., 10^{-2} or 10^{-3}) and then solve the above system numerically. For each x_3 value and for each exploration phase defined in Algorithm 3, we compute the worst-case evacuation time. We select the x_3 value that minimizes this worst-case time. All this numerical work is implemented in Matlab. We iterate over x_3 in the interval $[0, 2\pi - s + 1]$. The upper bound for x_3 stems from the case $x_1 = y = 0$. Indeed, notice that, for $s = 1$, Fast-Chord is exactly BSP when we set $x_1 = y = 0$. For the time parameter, namely t, we iterate in the interval $\left[0, 1 + \frac{x_1+x_2}{s} + \frac{x_3}{s+1}\right]$. Finally, we use a parametric representation of the disk, where the center O lies on coordinates $(0,0)$, to calculate the distance between the two robots.

By studying the numerical bounds we obtain via Fast-Chord, we state the following result, in comparison to the other two strategies studied in this paper. Below, let c_x stand for some constant that approximates $x \in \mathbb{R}$, i.e., $c_x \approx x$. That is, we use these c_x values because, due to lack of numerical accuracy, we are unable to determine the exact values for which the theorem below holds.

[1]The related source code is available at `https://github.com/yiannislamprou/FastDiskEvacuation`

Algorithm 4: Fast-Chord for Slow robot

Traverse line segment OC // $t = 0 \dots 1$
Traverse arc $\widehat{CD}$ // $t = 1 \dots 1 + y$
Traverse arc $\widehat{DB}$ till you meet Fast // $t = 1 \dots 1 + y + x_3/(s+1)$

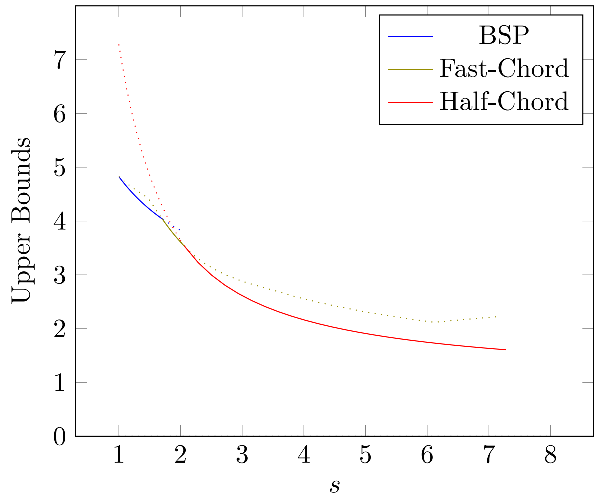


Figure 8: Comparison of upper bounds: solid lines signify the strongest upper bound among the three

Theorem 3. *Fast-Chord outperforms (generalized) Half-Chord for* $1 \le s \le c_{2.07}$*. It also outperforms Both-to-the-Same-Point for* $c_{1.71} \le s \le 2$*.*

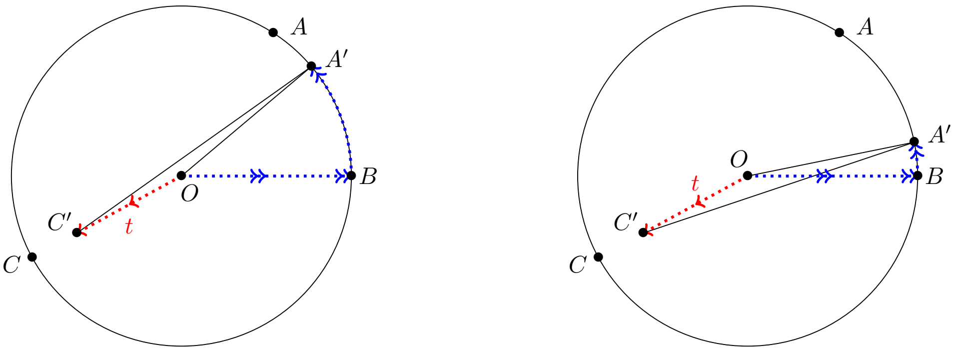
For a summary of numerical values leading to the above theorem, see Table 1. With respect to Fast-Chord and BSP, the improvement we gain in the interval $[c_{1.71}, 2]$ increases as s increases and reaches at most 6%, realized when $s = 2$. With respect to Half-Chord, it improves over BSP for $s \in [c_{1.86}, 2]$, but is still weaker than Fast-Chord for $s \le c_{2.07}$. In this interval, i.e., $s \in [c_{1.86}, c_{2.07}]$, Fast-Chord improves over Half-Chord by approximately at most 3%, realized when $s = 1.86$. Put together, in Figure 8, we demonstrate the comparison between the three upper bound strategies.

To conclude this section, we hereby provide the details of the parametric distance calculations we use to validate (up to a certain extent of numerical accuracy) the result in Theorem 3. Suppose the unit disk is embedded on a two-dimensional Cartesian coordinate system with the center O lying at point $(0, 0)$. Below, let $Fast_x$ and $Fast_y$ stand for the (x, y) coordinates of Fast's position and similarly $Slow_x$ and $Slow_y$ for Slow. The distances between the two robots at any given time are as follows (using the phases given in Algorithm 3):

Phase I. At time $t \in \left(\frac{1}{s}, 1\right]$, Fast has searched an $st-1$ part of $\widehat{BA}$ (until point A'), while Slow has traversed a t part of OC (until point C'); see Figure 9. Their distance is given by applying the cosine law in $\triangle A'OC'$. We compute the *in-triangle* angle $\measuredangle A'OC'$. In case that $\widehat{A'C'} \le \pi$ (case i), then $\measuredangle A'OC' = \widehat{BC} - \widehat{BA'} = s-1+x_1-(st-1) = s(1-t)+x_1$. Otherwise, if $\widehat{A'C'} > \pi$ (case ii), then $\measuredangle A'OC' = 2\pi - \widehat{A'A} - \widehat{AC} = 2\pi - (s-1-(st-1)) - x_1 = 2\pi - s(1-t) - x_1$. In either case, $|A'C'| = \sqrt{|OA'|^2 + |OC'|^2 - 2|OA'||OC'|\cos(\measuredangle A'OC')} = \sqrt{1 + t^2 - 2t\cos(s(1-t)+x_1)}$, since $\cos(2\pi - x) = \cos(x)$ for any x.

Phase IIa. At time $t \in \left(1, 1 + \frac{x_1}{s}\right]$, both robots are traversing their respective arcs in counterclockwise fashion. Their positions are the following:

$$(Fast_x, Fast_y) = \left(\cos\left(s\left(t - \frac{1}{s}\right)\right),\ \sin\left(s\left(t - \frac{1}{s}\right)\right)\right)$$



(a) Case (i)

(b) Case (ii)

Figure 9: Fast-Chord: Exit During Phase I

$$(Slow_x, Slow_y) = (\cos(s-1+x_1+t-1),\ \sin(s-1+x_1+t-1))$$

where we take into account the initial timestep when they begin traversing their corresponding arcs and the starting position of Slow's arc. Their distance is calculated in the Euclidean norm with the formula $\sqrt{(Fast_x - Slow_x)^2 + (Fast_y - Slow_y)^2}$.

Phase IIb. At time $t \in (1+\frac{x_1}{s}, 1+\frac{x_1+x_2}{s}]$, while Slow continues on the same arc and so its coordinates remain the same as in Phase IIa, Fast is now traversing the CB chord of length x_2, where $C = (x_C, y_C)$ and $B = (x_B, y_B)$ in the coordinate system. If we consider a parametric form of a vector directed from C to B such that at time $1+\frac{x_1}{s}$ Fast lies on C and starts moving at speed s toward B, then at time t, Fast lies at:

$$(Fast_x, Fast_y) = \left(x_C + s\frac{t-1-\frac{x_1}{s}}{x_2}(x_B - x_C),\ y_C + s\frac{t-1-\frac{x_1}{s}}{x_2}(y_B - y_C) \right).$$

By plugging in the proper values for x_B, x_C, y_B, y_C dictated by Fast-Chord, i.e., $C = (\cos(s-1+x_1), \sin(s-1+x_1))$ and $B = (\cos(0), \sin(0)) = (1, 0)$, we result to $(Fast_x, Fast_y)$ being equal to:

$$\left(\cos(s-1+x_1) + s\frac{t-1-\frac{x_1}{s}}{x_2}(1-\cos(s-1+x_1)),\ \sin(s-1+x_1) + s\frac{t-1-\frac{x_1}{s}}{x_2}(-\sin(s-1+x_1)) \right).$$

The Slow to Fast distance is again calculated by $\sqrt{(Fast_x - Slow_x)^2 + (Fast_y - Slow_y)^2}$.

Phase IIc. Again, Slow is always on the same motion and its corresponding parametric coordinates do not need to change. Fast, on the other hand, commences a clockwise traversal on $\widehat{BD}$ from position 2π with speed s, after time step $1+\frac{x_1+x_2}{s}$, leading to the following position coordinates at time $t \geq 1+\frac{x_1+x_2}{s}$.

$$\begin{aligned}(Fast_x, Fast_y) = &\ \left(\cos\left(2\pi - s\left(t-1-\tfrac{x_1+x_2}{s}\right)\right),\ \sin\left(2\pi - s\left(t-1-\tfrac{x_1+x_2}{s}\right)\right)\right) \\ = &\ \left(\cos\left(s\left(t-1-\tfrac{x_1+x_2}{s}\right)\right),\ -\sin\left(s\left(t-1-\tfrac{x_1+x_2}{s}\right)\right)\right)\end{aligned}$$

The Slow to Fast distance is calculated by $\sqrt{(Fast_x - Slow_x)^2 + (Fast_y - Slow_y)^2}$, as before.

4 Lower Bounds

The main tool behind our lower bounds is the following lemma from [13], which considers the distance between two unexplored boundary points when the boundary has been explored partially. Therefore, it can be applied independently of any robot attributes, e.g., their different speeds.

Lemma 6 (Lemma 5 [13]). *Consider a boundary of a disk whose subset of total length $u+\epsilon > 0$ has not been explored for some $\epsilon > 0$ and $\pi \geq u > 0$. Then there exist two unexplored boundary points between which the distance along the boundary is at least u.*

s	Fast-Chord (Theorem 3)	Half-Chord (Theorem 1, Corollary 1)	Both-Same-Point (Theorem 2)
1.00	4.827	7.283	4.826
1.10	4.728	6.621	4.680
1.20	4.639	6.069	4.547
1.30	4.558	5.602	4.428
1.40	4.486	5.202	4.319
1.50	4.385	4.855	4.219
1.60	4.211	4.552	4.127
...	...	...	...
1.65	4.127	4.414	4.084
1.66	4.110	4.387	4.076
1.67	4.094	4.361	4.068
1.68	4.078	4.335	4.059
1.69	4.061	4.310	4.051
1.70	**4.045**	4.284	**4.043**
1.71	**4.029**	4.259	**4.035**
1.72	4.013	4.234	4.027
1.73	3.997	4.210	4.019
1.74	3.981	4.186	4.011
1.75	3.965	4.162	4.003
...	...	...	...
1.83	3.844	3.980	3.943
1.84	3.829	3.958	3.936
1.85	3.814	**3.937**	**3.928**
1.86	3.799	**3.916**	**3.921**
1.87	3.786	3.895	3.914
1.88	3.771	3.874	3.907
1.89	3.757	3.854	3.900
...	...	...	...
1.99	3.622	3.660	3.833
2.00	3.610	3.642	3.826
2.01	3.598	3.624	-
2.02	3.585	3.606	-
2.03	3.573	3.589	-
2.04	3.560	3.573	-
2.05	3.548	3.556	-
2.06	3.536	3.540	-
2.07	**3.523**	**3.524**	-
2.08	**3.512**	**3.509**	-
2.09	3.500	3.493	-
2.10	3.488	3.478	-
2.11	3.477	3.463	-
2.12	3.466	3.448	-
...	...	...	...
2.20	3.381	3.337	-
3.20	2.811	2.497	-
4.20	2.498	2.102	-
5.20	2.271	1.871	-
6.20	2.126	1.720	-

Table 1: Comparison of (numerical) Fast-Chord bounds and (symbolic) Half-Chord/BSP bounds. Fast-Chord outperforms Half-Chord for $s \in [1, 2.07]$, and BSP for $s \in [1.71, 2]$. Half-Chord outperforms BSP for $s \in [1.86, 2]$.

4.1 Fast Explores

Theorem 4. *Any FES-strategy takes (in the worst case) at least*

- $\frac{1+2\pi}{s}$ *time for any* $s \in [1,2]$ *and*
- $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$ *time for any* $s \geq 2$.

Proof. For any s, Fast needs at least $\frac{1+2\pi}{s}$ time to explore the whole boundary. We now show a better bound for $s \geq 2$. At time $\frac{1+a}{s}$ (where $a \geq 0$), Fast has explored at most an a part of the boundary. Then, if we consider the time $\frac{1+a-\epsilon}{s}$ (where $\epsilon > 0$), a $2\pi - (a-\epsilon) = 2\pi - a + \epsilon$ subset of the boundary has not been explored yet. We bound $a \in [\pi, 2\pi)$ such that $0 < 2\pi - a \leq \pi$ holds. We now apply Lemma 6 with $u = 2\pi - a$ and ϵ. Thence, there exist two unexplored boundary points between which the distance along the boundary is at least u. Let us now consider the perpendicular bisector of the chord connecting these two points. Depending on which side of the bisector Slow lies, an adversary may place the exit on the boundary point lying at the opposite side. The best case for Slow is to lie exactly on the point of the bisection. That is, Slow will have to cover a distance of at least $\frac{2\sin\left(\frac{u}{2}\right)}{2} = \sin\left(\frac{a}{2}\right)$, where $2\sin\left(\frac{u}{2}\right)$ is the chord length. In this case, the overall evacuation time is equal to $\frac{1+a}{s} + \sin\left(\frac{a}{2}\right)$ and for the best lower bound we compute

$$\max_{\pi \leq a < 2\pi} \left\{ \frac{1+a}{s} + \sin\left(\frac{a}{2}\right) \right\}.$$

The maximum is attained at $a' = 2\arccos\left(-\frac{2}{s}\right)$, and is defined only for $s \geq 2$. By plugging a' into the evacuation time function, we get a lower bound of $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$ (see appendix for details). Finally, notice that this bound is equal to $\frac{1+2\pi}{s}$ for $s = 2$ and greater than $\frac{1+2\pi}{s}$ for $s > 2$. □

4.2 Both Explore

The following lower bound is a result of applying Lemma 6 to obtain a generalization of the lower bound proved in [13]. The proof considers a timestep when both robots have searched some part of the boundary and lie on the opposite ends of a long chord. Then, an adversary either places the exit at the end opposite Fast or at the end being farthest to Slow; the latter leading to a chord bisection argument similar to the one used in Theorem 4.

Lemma 7. *Any BES-strategy takes (in the worst case) at least*

- $1 + \frac{2}{s}\sqrt{1-\frac{s^2}{(s+1)^2}} + \frac{-s+2\arccos\left(-\frac{s}{s+1}\right)+1}{s+1}$ *time for* $s \in [1,2)$,
- $1 + \sqrt{1-\frac{4}{(s+1)^2}} + \frac{-s+2\arccos\left(-\frac{2}{s+1}\right)+1}{s+1}$ *time for* $s \in [2, c_{4.84}]$ *(where* $c_{4.84} \approx 4.8406$*) and*
- $1 + \sin\left(\frac{s-1}{2}\right)$ *time for* $s \in (c_{4.84}, 2\pi+1)$.

Proof. At time 1, Fast has explored at most $s-1$ distance on the boundary, since it needs $\frac{1}{s}$ time to reach the boundary and in the remaining $\frac{s-1}{s}$ time it can traverse $s\frac{s-1}{s} = s-1$ distance. At time $1+y$, where $y \geq 0$ is a variable, Fast has explored at most an $s-1+sy$ part of the boundary and Slow has explored at most a y part of the boundary. We derive an upper bound for the variable y by noticing that the whole explored part can be strictly less than 2π (otherwise the exit has been found already): $s-1+(s+1)y < 2\pi \Rightarrow y < \frac{2\pi-s+1}{s+1}$. Notice that, since $y \geq 0$, we need $s < 2\pi+1$. Then, the unexplored part is strictly greater than $u := 2\pi - s + 1 - (s+1)y$. We apply the restriction that $u \leq \pi$, which holds for $y \geq \frac{\pi-s+1}{s+1}$. Moreover, $u > 0$ holds for any $s \geq 1$ given that $y < \frac{2\pi-s+1}{s+1}$.

Now, let us apply Lemma 6: There exist two unexplored points with arc distance at least u, which implies that the chord between them has length at least $2\sin\left(\frac{u}{2}\right) = 2\sin\left(\frac{2\pi-s-(s+1)y+1}{2}\right) = 2\sin\left(\frac{s+(s+1)y-1}{2}\right)$. An adversary can put the exit on any of the two endpoints. If Slow reaches an endpoint first (case I), then the exit is placed on the other side, such that Slow has to traverse the chord. If Fast reaches an endpoint first, then the exit is placed either on the other side (case II), meaning that Fast has to traverse the chord, or on the endpoint that lies the farthest from Slow's current position (case III), meaning that Slow has to traverse at least half the chord. Let $y_{min} = \max\{\frac{\pi-s+1}{s+1}, 0\}$ and $y_{max} = \frac{2\pi-s+1}{s+1}$. The worst-case evacuation time is

- $\max_{y\in[y_{min},y_{max})} \left\{1+y+2\sin\left(\frac{s+(s+1)y-1}{2}\right)\right\}$, when in case I,
- $\max_{y\in[y_{min},y_{max})} \left\{1+y+\frac{2}{s}\sin\left(\frac{s+(s+1)y-1}{2}\right)\right\}$, when in case II, and
- $\max_{y\in[y_{min},y_{max})} \left\{1+y+\sin\left(\frac{s+(s+1)y-1}{2}\right)\right\}$, when in case III.

It is easy to see that the best case scenario for an adversary is case I, i.e., when Slow has to traverse the chord. However, to have a conservative lower bound, we may assume that the robots will avoid case I. Then, for $s \in [1,2)$, the adversary may apply case II, since it provides a stronger bound than case III for these values of s, and place the exit on the opposite end of Fast's position so that Fast has to traverse the chord. Finally, for $s \geq 2$, the adversary will apply case III. The rest of the proof reduces to computing the maximum of these functions, with respect to y; for details see the proof addendum in the appendix. $\square$

The above lower bound, although it is strong for small values of s, becomes weak for larger values of s. This happens due to the fact that in the proof we consider only a specific moment of a both-explore strategy, where both robots have already explored some part of the boundary. Hence, there is a need to capture a lower bound for the case where Slow has not explored any part of the boundary yet. This is possible, since we can apply an FES lower bound idea when s is big enough.

Lemma 8. *Any BES-strategy takes (in the worst case) at least*

- $1+\sin\left(\frac{s-1}{2}\right)$ *time for* $s \in [\pi+1, c_{4.97})$*, where* $c_{4.97} \approx 4.9699$*, and*
- $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$ *time for* $s \geq c_{4.97}$.

Proof. One need only notice that, for $a = s-1 \geq \pi$, at time $\frac{1+a-\epsilon}{s}$, a $2\pi - a + \epsilon$ part of the boundary is yet unexplored, where $2\pi - a \leq \pi$. Moreover, Slow has not reached the boundary yet. Hence, we can view this as a fast-explores subcase. Then, after applying Lemma 6, we can compute $\max_{a\in[\pi,\min\{s-1,2\pi\}]}\left\{\frac{1+a}{s}+\sin\left(\frac{a}{2}\right)\right\}$. Due to the upper bound for a, the analysis provides a $1+\sin\left(\frac{s-1}{2}\right)$ lower bound for $s \in [\pi+1, c_{4.97})$ and the already seen $\frac{1+2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}$ bound for $s \geq c_{4.97}$ (see appendix). $\square$

The following theorem encompasses the above BES lower bounds in Lemmata 7 and 8 by taking the maximum for each value of s.

Theorem 5. *Any BES-strategy takes (in the worst case) at least*

- $1+\frac{2}{s}\sqrt{1-\frac{s^2}{(s+1)^2}} + \frac{-s+2\arccos\left(-\frac{s}{s+1}\right)+1}{s+1}$ *time for* $s \in [1,2)$,
- $1+\sqrt{1-\frac{4}{(s+1)^2}} + \frac{-s+2\arccos\left(-\frac{2}{s+1}\right)+1}{s+1}$ *for* $s \in [2, c_{4.84}]$,
- $1+\sin\left(\frac{s-1}{2}\right)$ *time for* $s \in (c_{4.84}, c_{4.97})$ *and*
- $\frac{1+2\arccos(-2/s)}{s} + \sqrt{1-\frac{4}{s^2}}$ *time for* $s \in [c_{4.97}, \infty)$.

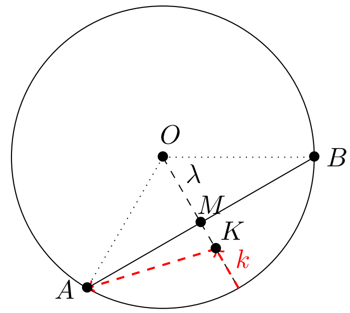


Figure 10: An Improved BES Lower Bound

4.3 Improved Both-Explore (Numerical) Lower Bound

We now obtain numerical values for a stronger BES lower bound by introducing a more complex argument building on the original BES lower bound proof given in Lemma 7. The main idea behind the improvement is to provide a better bound for case III when the adversary places the exit on the farthest endpoint from Slow's current position. Apparently, the best play for Slow is to lie exactly on the midpoint of the chord with the unexplored endpoints. Nevertheless, in order for Slow to be there, it needs to spend some of its time, originally destined for exploration, within the disk interior. We hereby examine the best possible scenario for Slow in terms of its distance from the midpoint following the above reasoning.

At time $1+y$, where $y \geq 0$ is a variable, Fast has explored at most an $s-1+sy$ part of the boundary and Slow has explored at most a y part of the boundary. Now suppose that Slow has spent k time, where $k \in [0, y]$, *not exploring* the boundary, i.e. moving within the disk interior. Notice that it takes at most $1 + \frac{2\pi-s+1}{s+1}$ time for the whole perimeter to be explored, when both robots are only exploring the boundary after time 1 (and not spending any time within the disk interior). Thence, we upper-bound $y < y_{max} := \frac{2\pi-s+1}{s+1}$. To lower-bound y, we restrict the unexplored part $u = 2\pi - s + 1 - (s+1)y + k \leq \pi$. That is, we get $y \geq y_{min} := \max\{\frac{\pi-s+1+k}{s+1}, 0\}$. Moreover, $u > 0$ is already covered by the aforementioned upper bound.

Now, we are ready to apply Lemma 6: There exist two unexplored points, namely A and B, at arc distance at least $2\pi - s + 1 - (s+1)y + k$, which implies that the chord between them has length at least $2\sin\left(\frac{2\pi-s+1-(s+1)y+k}{2}\right) = 2\sin\left(\frac{s-1+(s+1)y-k}{2}\right)$. An adversary could place the exit on any of the two endpoints. If Slow reaches an endpoint first (case I), then the exit is placed on the other side, such that Slow has to traverse the chord. If Fast reaches an endpoint first, then the exit is placed either on the other side (case II), meaning that Fast has to traverse the chord, or on the endpoint that lies the farthest from Slow's current position (case III). The worst-case evacuation time in cases I and II is given by

- $\max_{y\in[y_{min}, y_{max})}\left\{1 + y + 2\sin\left(\frac{s+(s+1)y-1-k}{2}\right)\right\}$, when in case I, and
- $\max_{y\in[y_{min}, y_{max})}\left\{1 + y + \frac{2}{s}\sin\left(\frac{s+(s+1)y-1-k}{2}\right)\right\}$, when in case II.

It is easy to see that the best case scenario, between these two, for an adversary is case I. However, to have a conservative lower bound, we may assume that the robots will avoid case I.

Let us now examine more carefully what happens in case III. For a depiction, see Figure 10. The ideal location for Slow is to lie exactly on the chord midpoint, say M (like in the proof of Lemma 7). Nevertheless, this may not be possible due to it only spending k time within the disk interior (after it first reaches the boundary). Let us consider the minimum distance from the chord midpoint to the boundary. This is exactly $1-\lambda$, where $\lambda = |OM|$ is the distance from the midpoint to the center of the disk. Notice that OM intesects AB *perpendicularly*, since M is the midpoint of chord AB. Using the Pythagorean theorem in $\triangle AMO$, we get $\lambda = \sqrt{1 - \sin^2\left(\frac{s-1+(s+1)y-k}{2}\right)}$. If we consider the case when $1 - \lambda > k$, then the ideal position for Slow is to lie k distance away from the boundary and on the extension of OM, i.e., on point K. From there, Slow can take a straight line to the exit, yielding a $\sqrt{\sin^2\left(\frac{s-1+(s+1)y-k}{2}\right) + (1-\lambda-k)^2}$ distance

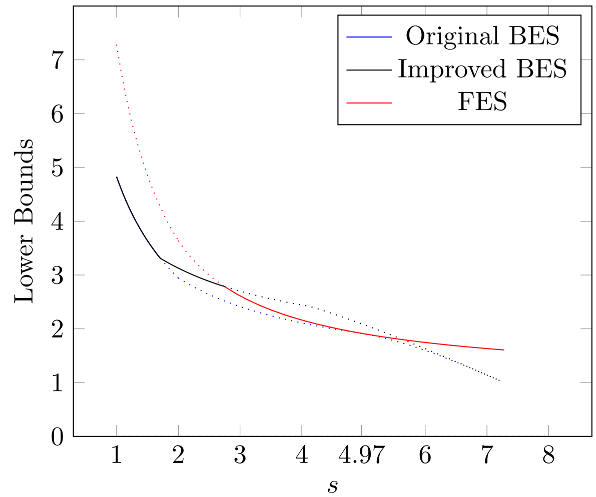


Figure 11: Comparison of lower bounds: solid lines signify the overall fast evacuation lower bound

again computed by the Pythagorean theorem, now in $\triangle AMK$. To conclude, Slow will try to minimize this straight line distance over k, while the adversary will select a case between II and III that maximizes the total distance. Overall, the optimization problem reduces to computing the following expression:

$$\max_{y\in[y_{min},y_{max})} \left\{ 1+y+\max\left\{ \begin{array}{l} \min\limits_{k\in[0,y]} \frac{2}{s}\sin\left(\frac{s-1+(s+1)y-k}{2}\right), \\ \min\limits_{k\in[0,y]} \sqrt{\sin^2\left(\frac{s-1+(s+1)y-k}{2}\right)+\max\left\{1-\lambda-k,0\right\}^2} \end{array} \right\} \right\} \tag{1}$$

Note that the above bound matches the one in Lemma 7 for $1-\lambda<k$.

Last but not least, we need also consider the case the adversary chooses to place the exit on the last boundary point to be explored. In the current setting, it takes at least $\frac{u}{s+1}=\frac{2\pi-s+1-(s+1)y+k}{s+1}$ extra time for both robots to explore the rest of the boundary, since Fast explores $s\frac{u}{s+1}$ while Slow explores $\frac{u}{s+1}$ for a total distance of u. Overall, we are looking to compute $\max\limits_{y\in[y_{min},y_{max})}\left\{1+y+\frac{2\pi-s+1-(s+1)y}{s+1}\right\}$, since Slow wishes to minimize k in this case. Due to the inherent complexity of the optimization problem (1), we compute *numerical* bounds[2]. The two $\min_{k\in[0,y]}$ expressions are computed and the maximum of them is chosen as the best-play scenario for an adversary. The computational work is done in Matlab, where we iterate over feasible values of y and k with a step of 10^{-3}. For Fast's speed s, we iterate with a step of 10^{-1}. The resulting bounds show that, for all $s\in[1,2\pi+1)$, this lower bound is greater or equal to the lower bound given in Lemma 7 with $k=0$ *always* selected as the minimizer.

Comparison of Lower Bounds. In Table 2, see also Figure 11, we present a summary of values for our established lower bounds. For the selected accuracy of speed s, we verify that for each $s\in[1,2\pi+1)$, the numerical values from this section are at least their corresponding values obtained by the original BES lower bound (in Lemma 7); see the first two columns of bounds. Moreover, for any $s\geq c_{1.71}\approx 1.71$, they are strictly stronger. In the "Max BES" column, we select the maximum among the three derived BES lower bounds. For the overall fast evacuation lower bound, for each value of s we select the minimum (weakest) lower bound between the (maximum) BES and FES ones as our overall lower bound. Improved BES is smaller than the FES lower bound (Theorem 4) for $s\leq c_{2.75}\approx 2.75$.

[2]The related source code is available at `https://github.com/yiannislamprou/FastDiskEvacuation`

s	Original BES (Lemma 7)	Improved BES (Section 4.3)	Fast-Like BES (Lemma 8)	Max BES (Theorem 5)	FES (Theorem 4)	Min(FES,BES)
1.00	4.826	4.826	-	4.826	7.283	4.826
1.20	4.258	4.258	-	4.258	6.069	4.258
1.40	3.822	3.822	-	3.822	5.202	3.822
1.60	3.473	3.473	-	3.473	4.552	3.473
1.69	3.337	3.337	-	3.337	4.310	3.337
1.70	**3.323**	**3.323**	-	3.323	4.284	3.323
1.71	**3.309**	**3.311**	-	3.311	4.259	3.311
1.72	**3.294**	**3.305**	-	3.305	4.234	3.305
1.73	3.280	3.298	-	3.298	4.210	3.298
1.80	3.186	3.250	-	3.250	4.046	3.250
2.00	2.946	3.128	-	3.128	3.642	3.128
2.20	2.809	3.021	-	3.021	3.337	3.021
2.40	2.691	2.926	-	2.926	3.099	2.926
2.60	2.587	2.842	-	2.842	2.907	2.842
2.72	2.531	2.796	-	2.796	2.808	2.796
2.73	2.526	2.793	-	**2.793**	2.800	**2.793**
2.74	2.521	2.789	-	**2.789**	2.792	**2.789**
2.75	2.517	2.785	-	**2.785**	**2.785**	**2.785**
2.76	2.513	2.782	-	2.782	**2.777**	**2.777**
2.77	2.508	2.778	-	2.778	**2.769**	**2.769**
2.80	2.495	2.767	-	2.767	2.747	2.747
3.00	2.413	2.700	-	2.700	2.612	2.612
3.20	2.340	2.638	-	2.638	2.497	2.497
3.40	2.274	2.583	-	2.583	2.397	2.397
3.60	2.214	2.532	-	2.532	2.309	2.309
3.80	2.159	2.485	-	2.485	2.232	2.232
4.00	2.109	2.443	-	2.443	2.163	2.163
4.20	2.064	2.393	2.000	2.393	2.102	2.102
4.40	2.022	2.320	1.992	2.320	2.046	2.046
4.60	1.983	2.243	1.974	2.243	1.996	1.996
4.80	1.947	2.163	1.946	2.163	1.951	1.951
5.00	1.909	2.081	1.909	2.081	1.909	1.909
5.20	1.863	1.995	1.871	1.995	1.871	1.871
5.40	1.808	1.907	1.836	1.907	1.836	1.836
5.60	1.746	1.817	1.804	1.817	1.804	1.804
5.80	1.675	1.725	1.774	1.774	1.774	1.774
6.00	1.598	1.631	1.746	1.746	1.746	1.746
6.20	1.516	1.535	1.720	1.720	1.720	1.720
6.40	1.427	1.438	1.696	1.696	1.696	1.696
6.60	1.335	1.340	1.674	1.674	1.674	1.674
6.80	1.239	1.241	1.653	1.653	1.653	1.653
7.00	1.141	1.141	1.633	1.633	1.633	1.633
7.20	1.042	1.042	1.614	1.614	1.614	1.614
7.26	1.012	1.012	1.609	1.609	1.609	1.609
7.28	1.002	1.002	1.607	1.607	1.607	1.607

Table 2: Comparison of derived numerical and theoretical lower bounds

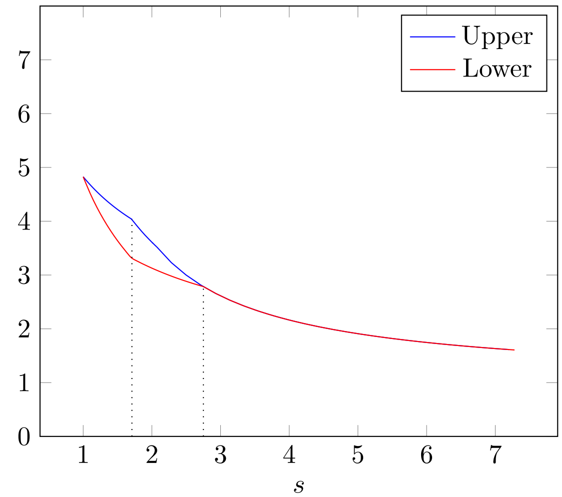


Figure 12: Dominant Lower vs Upper Bounds

5 Conclusions

By comparing the prevailing upper and lower bounds, we see, Figure 12, that Half-Chord (Theorem 1) is optimal for $s \geq c_{2.75}$, since the matching FES lower bound is the weakest in this interval (Table 2). On the other hand, for $s < c_{2.75}$ the ratio between the bounds is at most 1.22 (maximized when $s = c_{1.71}$), where the strategy changes from BSP to Fast-Chord. The best strategy to use is BSP when $s < c_{1.71}$, Fast-Chord when $c_{1.71} < s < c_{2.07}$ and Half-Chord for $s \geq c_{2.07}$; see Table 1.

Optimality for the case $1 < s < c_{2.75}$ remains open. In this gray area, the main difficulty is understanding when it becomes necessary to make the transition from a BES to an FES strategy. As indicated by our introduction of Fast-Chord, which outperforms BSP and Half-Chord in the interval $(c_{1.71}, c_{2.07})$, the potential strategies might need to become even more convoluted to capture the diminishing speed ratio.

Regarding future work on this topic, one could consider extending these results to a more-than-two-robots evacuation scenario. Moreover, the non-wireless case for two-robots fast evacuation seems to be an even more challenging open problem given that exact optimality is complex to obtain even for $s = 1$ ([18, 9]). Finally, other environments could be examined, e.g. polygonal ones [19], or more realistic robotic settings where the environment becomes more perplexing, e.g., including spatial obstacles or communication restrictions.

References

[1] S. Alpern, R. Fokkink, L. Gąsieniec, R. Lindelauf, V.S. Subrahmanian, Search theory, *Springer*, 2013.

[2] S. Alpern and S. Gal. The Theory of Search Games and Rendezvous. *Int. Series in Operations Research and Management Science*, number 55, Kluwer Academic Publishers, 2002.

[3] R.A. Baeza-Yates, J.C. Culberson, and G.J.E. Rawlins. Searching in the plane. *Inform. and Comput.* **106** (1993), pp. 234–252.

[4] A. Beck. On the linear search problem. *Naval Res. Logist.* **2** (1964), pp. 221–228.

[5] R. Bellman. Minimization problem. *Bull. AMS* **62** (1956), p. 270.

[6] R. Bellman. An optimal search problem. *SIAM Rev.* **5** (1963), p. 274.

[7] A. Bonato and R.J. Nowakowski. *The Game of Cops and Robbers on Graphs.* American Mathematical Society, 2011.

[8] P. Bose, J.-L. De Carufel, and S. Durocher. Revisiting the problem of searching a line. *ESA 2013*, pp. 205–216.

[9] S. Brandt, F. Laufenberg, Y. Lv, D. Stolz, R. Wattenhofer. Collaboration Without Communication: Evacuating Two Robots from a Disk. *CIAC 2017*, pp. 104–115.

[10] Y. Chevaleyre. Theoretical analysis of the multi-agent patrolling problem. *IAT 2004*, pp. 302–308.

[11] M. Chrobak, L. Gąsieniec, T. Gorry, R. Martin. Group search on the line. *SOFSEM 2015*, pp. 164–176.

[12] H. Chuangpishit, K. Georgiou, P. Sharma, Average Case - Worst Case Tradeoffs for Evacuating 2 Robots from the Disk in the Face-to-Face Model, *Accepted at ALGOSENSORS 2018*.

[13] J. Czyzowicz, L. Gąsieniec, T. Gorry, E. Kranakis, R. Martin, and D. Pająk, Evacuating robots from an unknown exit located on the boundary of a disc, *DISC 2014*, LNCS 8784, pp. 122–136, Springer, 2014.

[14] J. Czyzowicz, L. Gąsieniec, A. Kosowski, and E. Kranakis. Boundary patrolling by mobile agents with distinct maximal speeds. *ESA 2011*, pp. 701–712.

[15] J. Czyzowicz, K. Georgiou, M. Godon, E. Kranakis, D. Krizanc, W. Rytter, M. Wlodarczyk, Evacuation from a Disc in the Presence of a Faulty Robot, *SIROCCO 2017*, pp. 158–173.

[16] J. Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, S. Shende, God Save the Queen, *FUN 2018*.

[17] J. Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, S. Shende, Priority Evacuation from a Disk Using Mobile Robots, *SIROCCO 2018*.

[18] J. Czyzowicz, K. Georgiou, E. Kranakis, L. Narayanan, J. Opatrny, and B. Vogtenhuber, Evacuating robots from a disk using face-to-face communication, *CIAC 2015*, pp. 140–152.

[19] J. Czyzowicz, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. Shende, Wireless Autonomous Robot Evacuation from Equilateral Triangles and Squares, *Ad-hoc, Mobile, and Wireless Networks (ADHOC-NOW)*, pp. 181–194, Springer Cham, 2015.

[20] O. Feinerman, A. Korman, S. Kutten and Y. Rodeh, Fast Rendezvous on a cycle by agents with different speeds, *ICDCN 2014*, LNCS, vol. 8314, pp. 1-13, Springer, 2014.

[21] J. Czyzowicz, K. Georgiou, E. Kranakis, Group Search and Evacuation, In: P. Flocchini, G. Prencipe, N. Santoro (Eds.), Distributed Computing by Mobile Entities LNCS, Springer, vol. 11340, pp. 335–370, 2019.

[22] K. Georgiou, G. Karakostas, E. Kranakis, Search-and-Fetch with One Robot on a Disk, *ALGOSENSORS 2016*, pp. 80–94.

[23] K. Georgiou, G. Karakostas, E. Kranakis, Search-and-Fetch with 2 Robots on a Disk: Wireless and Face-to-Face Communication Models, *ICORES 2017*.

[24] B. Gluss. An alternative solution to the "lost at sea" problem. *Naval Research Logistics Quarterly*, **8** (1961), pp. 117–122.

[25] E. Huus and E. Kranakis, Rendezvous of Many Agents with Different Speeds in a Cycle, *ADHOC-NOW 2015*, LNCS 9143, pp. 195-209, Springer, 2015.

[26] A. Jeż, and J. Łopuzański. On the two-dimensional cow search problem. *Information Processing Letters* **131** (2009), pp. 543–547.

[27] M.-Y. Kao, J.H. Reif, and S.R. Tate. Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem. *Inform. and Comput.* **131** (1996), pp. 63–80.

[28] E. Kranakis, D. Krizanc, and E. Markou. *The Mobile Agent Rendezvous Problem in the Ring.* Morgan & Claypool Publishers, 2010.

[29] E. Kranakis, D. Krizanc, S. Rajsbaum. Mobile agent rendezvous: A survey. *SIROCCO 2006*, pp. 1–9.

[30] T.D. Parsons. Pursuit-evasion in a graph, in *Theory and Applications of Graphs: Proc., Michigan May 11–15*, pp. 426–441, 1976.

[31] D. Pattanayak, H. Ramesh, P. S. Mandal, S. Schmid, Evacuating Two Robots from Two Unknown Exits on the Perimeter of a Disk. *ICDCN 2018*, article no. 20, 2018.

[32] V. Yanovski, I.A Wagner, and A.M. Bruckstein. A distributed ant algorithm for efficiently patrolling a network. *Algorithmica* **37** (2003), pp. 165–186.

A Missing Details from Proofs in Section 3

Proof Addendum for Lemma 4. Recall that we wish to compute the quantity $\max_{0\leq a\leq s-1}\left\{\frac{a+1}{s}+b\right\}$, where $b=\sqrt{1+\left(\frac{a+1}{s}\right)^2-2\cdot\frac{a+1}{s}\cos(a)}$. Let $f(a,s)=\frac{a+1}{s}+b$. Then, we compute the derivative

$$\frac{\partial}{\partial a}f(a,s)=\frac{1}{s}+\frac{\frac{2(a+1)}{s^2}+\frac{2(a+1)\sin(a)}{s}-\frac{2\cos(a)}{s}}{2\sqrt{1+\left(\frac{a+1}{s}\right)^2-2\frac{a+1}{s}\cos(a)}}\geq 0$$

for any $a\leq s-1$. Consequently, $f(a,s)$ is a non-decreasing function of a in this interval meaning that the maximum is attained on $a=s-1$. This results to a worst-case evacuation time of

$$f(s-1,s)=\frac{s-1+1}{s}+\sqrt{1+\left(\frac{s-1+1}{s}\right)^2-2\frac{s-1+1}{s}\cos(s-1)}=1+\sqrt{2-2\cos(s-1)}.$$

□

Proof Addendum for Lemma 5. Recall that we seek to compute $\max_{0\leq d\leq\frac{\pi-s+1}{s+1}}\left\{1+d+2\sin\left(\frac{d(s+1)+s-1}{2}\right)\right\}$, where $s\in[1,2]$. We denote by $g(d,s)$ the function to be maximized. We compute

$$\frac{\partial}{\partial d}g(d,s)=1+(s+1)\cos\left(\frac{(s+1)d+s-1}{2}\right)$$

In case $angle(d,s)\leq\pi$, which implies $d\in[0,\frac{\pi-s+1}{s+1}]$, it follows $\frac{\partial}{\partial d}g(d,s)\geq 0$. The latter holds due to the fact that $s+1\geq 0$ and $\cos\left(\frac{(s+1)d+s-1}{2}\right)\geq\cos(\pi/2)=0$, since $(s+1)d+s-1\leq\pi$ and $\cos(\cdot)$ is decreasing in $[0,\pi/2]$. Hence, the maximum is attained at $d=\frac{\pi-s+1}{s+1}$ for a worst-case time of

$$\begin{aligned}
g\left(\tfrac{\pi-s+1}{s+1},s\right) &= 1+\tfrac{\pi-s+1}{s+1}+2\sin\left(\frac{(s+1)\frac{\pi-s+1}{s+1}+s-1}{2}\right)\\
&= 1+\tfrac{\pi-s+1}{s+1}+2\sin(\pi/2)\\
&= 3+\tfrac{\pi-s+1}{s+1}\\
&= \tfrac{2s+\pi+4}{s+1}.
\end{aligned}$$

In case $\pi<angle(d,s)<2\pi$, let us examine the derivative of $g(d,s)$. The family of roots for $\frac{\partial g(d,s)}{\partial d}=0$ is $d=\frac{4\pi n\pm 2\cdot\arccos(-1/(s+1))-s+1}{s+1}$. A local maximum is attained for $d'=\frac{2\cdot\arccos(-1/(s+1))-s+1}{s+1}$ since d' is the only root lying within (d_{min},d_{max}), see Figure 13, and, since $\sin(\arccos(x))=\sqrt{1-x^2}$ for any x, we get

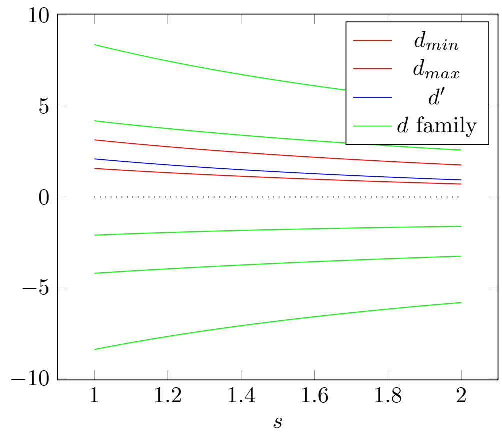


Figure 13: d' is the only root within $[d_{min}, d_{max}]$

$$
\begin{aligned}
\frac{\partial^2 g(d,s)}{\partial d^2}\Big|_{d=d'} \quad & -\tfrac{1}{2}(s+1)^2 \sin\left(\frac{s+(s+1)\frac{2\arccos\left(-\frac{1}{s+1}\right)-s+1}{s+1}-1}{2}\right) \\
&= -\tfrac{1}{2}(s+1)^2 \sin\left(\arccos\left(-\tfrac{1}{s+1}\right)\right) \\
&= -\tfrac{1}{2}(s+1)^2 \sqrt{1-\left(-\tfrac{1}{s+1}\right)^2} \\
&< 0
\end{aligned}
$$

Finally, we compute

$$
\begin{aligned}
g(d', s) \quad &= 1 + \tfrac{2\cdot\arccos(-1/(s+1))-s+1}{s+1} + 2\sin\left(\frac{s+(s+1)\frac{2\arccos\left(-\frac{1}{s+1}\right)-s+1}{s+1}-1}{2}\right) \\
&= 1 + \tfrac{2\cdot\arccos(-1/(s+1))-s+1}{s+1} + 2\sin\left(\arccos\left(-\tfrac{1}{s+1}\right)\right) \\
&= 1 + \tfrac{2\cdot\arccos(-1/(s+1))-s+1}{s+1} + 2\sqrt{1-\left(-\tfrac{1}{s+1}\right)^2}
\end{aligned}
$$

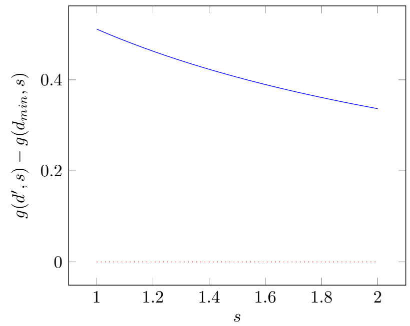
which is a globally optimal value, since $g(d', s) > g(d_{min}, s) = \frac{2s+\pi+4}{s+1}$ and $g(d', s) > g(d_{max}, s) = \frac{2\pi+2}{s+1}$ for any $s \in [1, 2]$; see Figure 14.

□

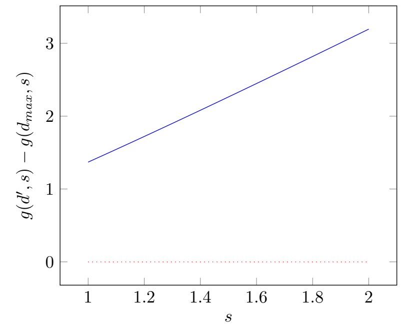
B Missing Details from Proofs in Section 4

Proof Addendum for Theorem 4. Recall that we seek to compute $\max\limits_{\pi \le a < 2\pi} \left\{\frac{1+a}{s} + \sin\left(\frac{a}{2}\right)\right\}$. The first partial derivative is equal to $\frac{\partial f(s,a)}{\partial a} = \frac{1}{s} + \frac{1}{2}\cos\left(\frac{a}{2}\right)$ and the family of solutions to $\frac{\partial f(s,a)}{\partial a} = 0$ is of the form:

$$\left\{4\pi n \pm 2\arccos\left(-\frac{2}{s}\right) : n \in \mathbb{Z}\right\}$$



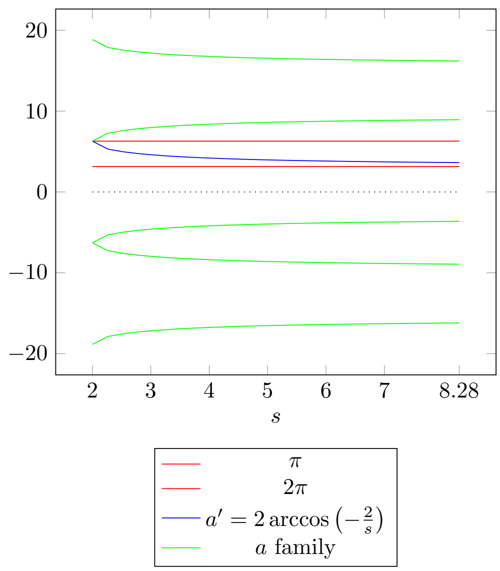
(a) $g(d', s) > g\left(\frac{\pi+1-s}{s+1}, s\right)$ calculation



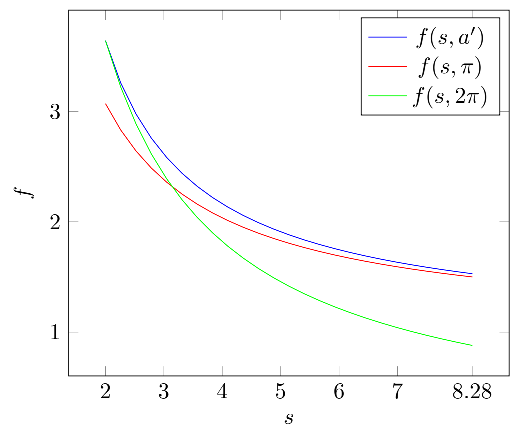
(b) $g(d', s) > g\left(\frac{2\pi+1-s}{s+1}, s\right)$ calculation

Figure 14: d' is optimum for $s \in [1, 2]$

The only solution which is included in the interval $[\pi, 2\pi)$ is $a' = 2\arccos\left(-\frac{2}{s}\right)$, which it is defined *only for* $s \geq 2$ due to the $\arccos(\cdot)$ function; see Figure 15a. Moreover, a' is a local maximum, since



(a) a' is the only root within $[\pi, 2\pi)$



(b) Comparison of potential global maxima in $[\pi, 2\pi)$

Figure 15: Accompanying figures for proof calculations (Theorem 4)

$$\frac{\partial^2 f(s,a)}{\partial a^2}\Big|_{a=a'} = -\frac{1}{4}\sin\left(\frac{2\arccos\left(-\frac{2}{s}\right)}{2}\right) = -\frac{1}{4}\sqrt{1-\frac{4}{s^2}} < 0$$

for any $s \geq 2$. It then suffices to compare $f(s, a')$ to $f(s, \pi) = 1 + \frac{1+\pi}{s}$ and $f(s, 2\pi) = \frac{1+2\pi}{s}$ to prove global optimality. Indeed, one can verify $f(s, a') \geq f(s, \pi)$ and $f(s, a') \geq f(s, 2\pi)$ for any $s \geq 2$; see Figure 15b for an example depiction for some small values of s. The lower bound is

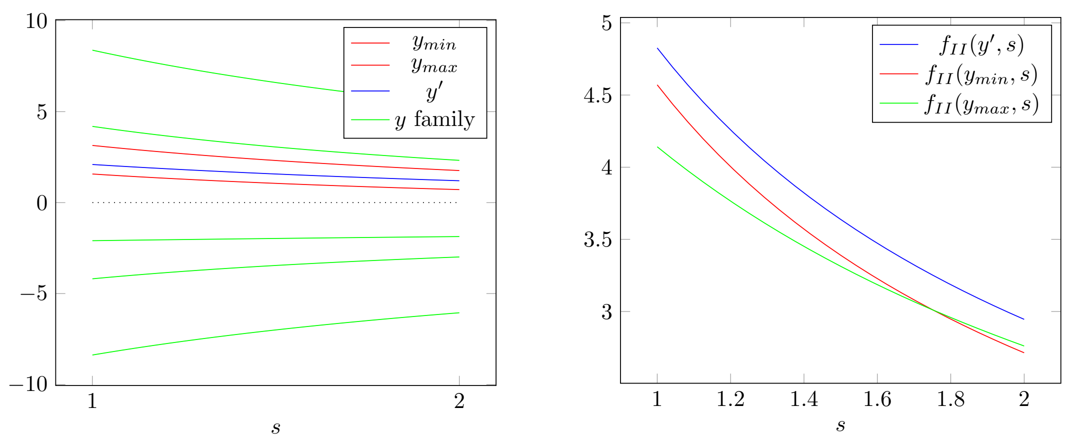
$$f(s, a') = \frac{1 + 2\arccos\left(-\frac{2}{s}\right)}{s} + \sin\left(\frac{2\arccos\left(-\frac{2}{s}\right)}{2}\right) = \frac{1 + 2\arccos\left(-\frac{2}{s}\right)}{s} + \sqrt{1-\frac{4}{s^2}}.$$

□

Proof Addendum for Lemma 7. Let $f_{II}(y,s) = 1 + y + \frac{2}{s}\sin\left(\frac{(s+1)y+s-1}{2}\right)$ be the function arising from case II. We only analyze the function for $s \in [1,2)$, since for $s \geq 2$ it is easy to see that case III provides a stronger lower bound. We compute

$$\frac{\partial}{\partial y}\left(1 + y + \frac{2}{s}\sin\left(\frac{(s+1)y+s-1}{2}\right)\right) = 1 + \frac{(s+1)\cos\left(\frac{(s+1)y+s-1}{2}\right)}{s}$$

which gives the family of roots $y = \frac{4\pi n \pm 2\arccos\left(-\frac{s}{s+1}\right) - s + 1}{s+1}$ $(n \in \mathbb{Z})$. The only root (and thus potential maximum of the function) that lies within $[y_{min}, y_{max}]$ is $y' = \frac{2\arccos\left(-\frac{s}{s+1}\right)-s+1}{s+1}$, see Figure 16a. Moreover, we can see that $y' > 0$ for $s \in [1,2)$ as needed, since y' represents distance.



(a) y' is the only root within the y_{min}, y_{max} bounds

(b) y' is the global maximum for $s \in [1,2)$

Figure 16: Helpful plots for the calculations in Case II

We demonstrate concavity at y'

$$\begin{aligned}
\frac{\partial^2 f_{II}(y,s)}{\partial y^2}\Big|_{y=y'} &= \frac{\partial}{\partial y}\left(1 + \frac{(s+1)\cos(\left(\frac{(s+1)y+s-1}{2}\right)}{s}\right)\Big|_{y=y'} \\
&= -\frac{(s+1)^2\sin\left(\frac{(s+1)y+s-1}{2}\right)}{2s}\Big|_{y=y'} \\
&= -\frac{(s+1)^2\sqrt{1-\frac{s^2}{(s+1)^2}}}{2s} < 0
\end{aligned}$$

and compute the value of f_{II} at y'

$$\begin{aligned}
f_{II}(y', s) &= f_{II}\left(\frac{2\arccos\left(-\frac{s}{s+1}\right)-s+1}{s+1}, s\right) \\
&= 1 + \frac{2\arccos\left(-\frac{s}{s+1}\right)-s+1}{s+1} + \frac{2}{s}\sin\left(\arccos\left(-\frac{s}{s+1}\right)\right) \\
&= 1 + \frac{2\arccos\left(-\frac{s}{s+1}\right)-s+1}{s+1} + \frac{2\sqrt{1-\frac{s^2}{(s+1)^2}}}{s}
\end{aligned}$$

Finally, we compute the values at the interval endpoints

$$f_{II}(y_{min}, s) = f_{II}\left(\frac{\pi+1-s}{s+1}, s\right) = 1 + \frac{\pi+1-s}{s+1} + \frac{2}{s}\sin(\pi/2) = \frac{\pi s + 4s + 2}{s(s+1)}$$

$$f_{II}(y_{max}, s) = f_{II}\left(\frac{2\pi+1-s}{s+1}, s\right) = 1 + \frac{2\pi+1-s}{s+1} + \frac{2}{s}\sin(\pi) = \frac{2\pi+2}{s+1}$$

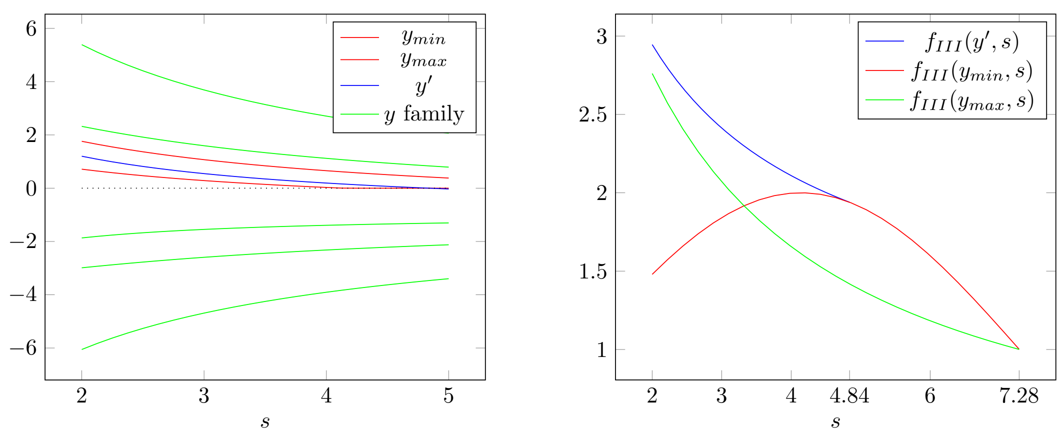
It suffices to verify they are always less to $f_{II}(y', s)$ for $s \in [1, 2]$, see Figure 16b.

Moving forward, let $f_{III}(y, s) = 1 + y + \sin\left(\frac{(s+1)y+s-1}{2}\right)$ stand for the function to be maximized arising from Case III. We follow the same steps as before.

$$\frac{\partial f_{III}(y, s)}{\partial y} = 1 + \frac{(s+1)\cos\left(\frac{(s+1)y+s-1}{2}\right)}{2}$$

gives the family of roots $y = \frac{4\pi n \pm 2\arccos\left(\frac{2}{s+1}\right) - s + 1}{s+1} \quad (n \in \mathbb{Z})$.

The only root (and thus potential maximum of the function) that lies within $[y_{min}, y_{max}]$ is $y' = \frac{2\arccos\left(\frac{2}{s+1}\right) - s + 1}{s+1}$; see Figure 17a. Moreover, we can see that $y' > 0$ holds *only* for $s \in [2, c_{4.84})$, where $c_{4.84} \approx 4.8406$.



(a) y' is the only root within the y_{min}, y_{max} bounds (b) y' is the global maximum for $s \in [2, c_{4.84})$

Figure 17: Helpful plots for the calculations in Case III

We demonstrate concavity at y'

$$\begin{aligned}
\frac{\partial^2 f_{III}(y,s)}{\partial y^2}\,\Big|_{y=y'} &= \frac{\partial}{\partial y}\left(1 + \frac{(s+1)\cos\left(\frac{(s+1)y+s-1}{2}\right)}{2}\right)\Big|_{y=y'} \\
&= -\frac{(s+1)^2\sin\left(\frac{(s+1)y+s-1}{2}\right)}{4}\,\Big|_{y=y'} \\
&= -\frac{(s+1)^2\sqrt{1-\frac{4}{(s+1)^2}}}{4} < 0
\end{aligned}$$

and compute the value of f_{III} at y'

$$
\begin{aligned}
f_{III}(y', s) &= f_{III}\left(\frac{2\arccos\left(\frac{2}{s+1}\right)-s+1}{s+1}, s\right) \\
&= 1 + \frac{2\arccos\left(\frac{2}{s+1}\right)-s+1}{s+1} + \sin\left(\arccos\left(\frac{2}{s+1}\right)\right) \\
&= 1 + \frac{2\arccos\left(\frac{2}{s+1}\right)-s+1}{s+1} + \sqrt{1-\frac{4}{(s+1)^2}}
\end{aligned}
$$

Finally, we compute $f_{III}(y_{min}, s)$, $f_{III}(y_{max}, s)$, which are less than $f_{III}(y', s)$ for $s \in [2, c_{4.84})$, where $c_{4.84} \approx 4.84$, see Fig. 17b.

$$
f_{III}(y_{min}, s) = f_{III}(0, s) = 1 + \sin\left(\frac{s-1}{2}\right)
$$

$$
f_{III}(y_{max}, s) = f_{III}\left(\frac{2\pi+1-s}{s+1}, s\right) = 1 + \frac{2\pi+1-s}{s+1} + \sin(\pi) = \frac{2\pi+2}{s+1}
$$

For the case $s \geq c_{4.84}$, we need only consider the endpoints of the $[y_{min}, y_{max}]$ interval as potential maxima: it holds $f_{III}(y_{min}, s) \geq f_{III}(y_{max}, s)$ for $s \in [c_{4.84}, 2\pi+1)$. $\square$

Proof Addendum for Lemma 8. Let $f(a, s) = \frac{1+a}{s} + \sin\left(\frac{a}{2}\right)$ be the emerging function that needs to be maximized for $a \in [\pi, \min\{s-1, 2\pi\}]$. This function is already analyzed in the proof of Theorem 4. Nevertheless, we now need to reconsider it, since the underlying domain depends on s. For $s \geq 2\pi+1$, $\min\{s-1, 2\pi\} = 2\pi$ and so the analysis proceeds as before yielding a lower bound of $\sqrt{1-\frac{4}{s^2}} + \frac{1+2\arccos(-2/s)}{s}$. Let us now consider $s \in [\pi+1, 2\pi+1)$. The selected derivative root is again $a' = 2\arccos\left(-\frac{2}{s}\right)$. Nonetheless, one ought to notice that $2\arccos\left(-\frac{2}{s}\right) \leq s-1$ *only for* $s \geq c_{4.97}$, where $c_{4.97} \simeq 4.9699$. Now, let us compare $f(a', s)$ to $f(\pi, s)$ and $f(s-1, s)$ (i.e. the values at the endpoints of the interval). We get $f(a', s) = \sqrt{1-\frac{4}{s^2}} + \frac{1+2\arccos(-2/s)}{s}$ as before, $f(\pi+1, s) = 1 + \frac{1+\pi}{s}$ and $f(s-1, s) = 1 + \sin\left(\frac{s-1}{2}\right)$.

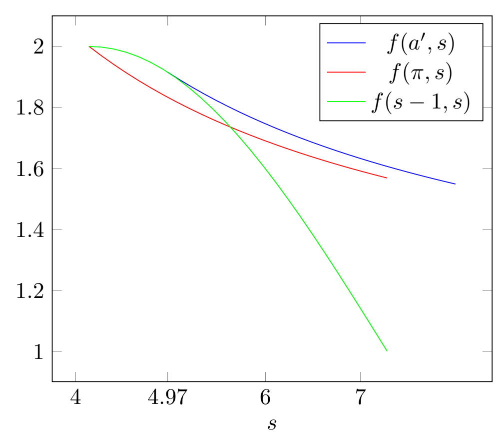


Figure 18: Comparison of potential maxima in proof of Lemma 8

One can notice, see Figure 18, that $f(a', s)$ prevails for $s \geq c_{4.97}$, while $f(s-1, s)$ is greater to $f(\pi, s)$ for $s \in (\pi+1, c_{4.97})$. $\square$